

Bluetooth® Enhanced Data Rate Digital Standard for R&S® Signal Generators Operating Manual



1171.6496.12 – 10

This document describes the following software options:

- R&S®AMU-K60/-K260
1403.0353.02, 1403.0401.02
- R&S®SMATE-K60
1404.8412.02
- R&S®SMBV-K60/-K260
1415.8477.xx, 1415.8483.xx
- R&S®SMJ-K60/-K260
1409.2858.02, 1409.2906.02
- R&S®SMU-K60/-K260
1408.7962.02, 1408.8017.02
- R&S®SMW-K260
1413.5287.02
- R&S®AFQ-K260
1401.5758.02
- R&S®SFU-K260
2115.2389.02

© 2013 Rohde & Schwarz GmbH & Co. KG

Mühldorfstr. 15, 81671 München, Germany

Phone: +49 89 41 29 - 0

Fax: +49 89 41 29 12 164

E-mail: info@rohde-schwarz.com

Internet: www.rohde-schwarz.com

Subject to change – Data without tolerance limits is not binding.

R&S® is a registered trademark of Rohde & Schwarz GmbH & Co. KG.

The *Bluetooth*® word mark and logos are owned by Bluetooth SIG, Inc. and any use of such marks by Rohde & Schwarz is under license.

Trade names are trademarks of the owners.

The following abbreviations are used throughout this manual: R&S®AMU200A is abbreviated as R&S AMU, R&S®SMATE200A is abbreviated as R&S SMATE, R&S®SMBV100A is abbreviated as R&S SMBV, R&S®SMJ100A is abbreviated as R&S SMJ, R&S®SMU200A is abbreviated as R&S SMU, R&S®SMW200A is abbreviated as R&S SMW, R&S®WinIQSIM2™ is abbreviated as R&S WinIQSIM2

Contents

1	Preface.....	5
1.1	Documentation Overview.....	5
1.2	Typographical Conventions.....	6
2	Introduction.....	7
2.1	Bluetooth Transport Modes.....	7
2.2	Bluetooth Packet Types.....	8
2.2.1	ACL packets.....	8
2.2.2	SCO and eSCO packets.....	9
2.2.3	Link control packets for ACL, SCO, eSCO transport modes.....	10
2.3	Packet Structure and Fields.....	11
2.3.1	Access code.....	11
2.3.2	Header.....	11
2.3.3	Payload format.....	12
2.4	Bluetooth Modulation Schemes.....	13
3	User Interface.....	15
3.1	General Settings for Bluetooth Signals.....	16
3.2	Bluetooth Basic Rate + EDR.....	20
3.3	Packet Configuration - Bluetooth Basic Rate + EDR.....	21
3.4	Bluetooth Low Energy.....	25
3.5	Event / Frame Configuration - Bluetooth LE.....	29
3.5.1	Advertising Event / Frame Configuration Settings.....	31
3.5.2	Data Event Connection Settings.....	32
3.5.3	Channel Table Settings.....	34
3.6	Packet Configuration - Bluetooth LE.....	35
3.6.1	Packet Configuration Settings.....	36
3.6.2	Test Packet Configuration Settings.....	44
3.7	Dirty Transmitter Test.....	45
3.8	Filter/Clipping Settings.....	48
3.8.1	Filter Settings.....	49
3.8.2	Modulation Settings.....	50
3.8.3	Clipping Settings.....	51

3.9	Power Ramping Settings.....	52
3.10	Trigger/Marker/Clock Settings.....	53
3.10.1	Trigger Settings.....	54
3.10.2	Marker Mode.....	59
3.10.3	Marker Delay.....	59
3.10.4	Clock Settings.....	60
3.10.5	Global Settings.....	62
4	Remote-Control Commands.....	63
4.1	General Commands.....	64
4.2	Filter/Clipping Settings.....	70
4.3	Trigger Settings.....	76
4.4	Marker Settings.....	82
4.5	Clock Settings.....	86
4.6	Power Ramping.....	89
4.7	Packet Configuration Setting.....	90
4.8	Dirty Transmitter Test.....	99
4.9	LE Commands.....	104
	List of Commands.....	127
	Index.....	131

1 Preface

1.1 Documentation Overview

The user documentation for the R&S Signal Generator consists of the following parts:

- Online Help system on the instrument,
- "Quick Start Guide" printed manual,
- Documentation CD-ROM with:
 - Online help system (*.chm) as a standalone help,
 - Operating Manuals for base unit and options,
 - Service Manual,
 - Data sheet and specifications,
 - Links to useful sites on the R&S internet.

Online Help

The Online Help is embedded in the instrument's firmware. It offers quick, context-sensitive access to the complete information needed for operation and programming. The online help contains help on operating the R&S Signal Generator and all available options.

Quick Start Guide

The Quick Start Guide is delivered with the instrument in printed form and in PDF format on the Documentation CD-ROM. It provides the information needed to set up and start working with the instrument. Basic operations and an example of setup are described. The manual includes also general information, e.g., Safety Instructions.

Operating Manuals

The Operating Manuals are a supplement to the Quick Start Guide. Operating Manuals are provided for the base unit and each additional (software) option.

These manuals are available in PDF format - in printable form - on the Documentation CD-ROM delivered with the instrument. In the Operating Manual for the base unit, all instrument functions are described in detail. Furthermore, it provides an introduction to remote control and a complete description of the remote control commands with programming examples. Information on maintenance, instrument interfaces and error messages is also given.

In the individual option manuals, the specific instrument functions of the option are described in detail. For additional information on default settings and parameters, refer to the data sheets. Basic information on operating the R&S Signal Generator is not included in the option manuals.

Service Manual

The Service Manual is available in PDF format - in printable form - on the Documentation CD-ROM delivered with the instrument. It describes how to check compliance with rated specifications, on instrument function, repair, troubleshooting and fault elimination. It contains all information required for repairing the instrument by the replacement of modules.

This manual can also be ordered in printed form (see ordering information in the data sheet).

Release Notes

The release notes describe new and modified functions, eliminated problems, and last minute changes to the documentation. The corresponding firmware version is indicated on the title page of the release notes. The current release notes are provided in the Internet.

Web Helps

Web helps are provided for the base unit and each additional (software) option. The content of the web helps correspond to the user manuals for the latest product versions.

The web help is an additional file format that offers quick online access. They are not intended to be downloaded but rather to access the required information directly from the R&S website.

Web helps are available at the R&S website, on the R&S Signal Generator product page at the "Downloads > Web Help" area.

1.2 Typographical Conventions

The following text markers are used throughout this documentation:

Convention	Description
"Graphical user interface elements"	All names of graphical user interface elements on the screen, such as dialog boxes, menus, options, buttons, and softkeys are enclosed by quotation marks.
KEYS	Key names are written in capital letters.
File names, commands, program code	File names, commands, coding samples and screen output are distinguished by their font.
<i>Input</i>	Input to be entered by the user is displayed in italics.
Links	Links that you can click are displayed in blue font.
"References"	References to other parts of the documentation are enclosed by quotation marks.

2 Introduction

The R&S Signal Generator provides you with ability to generate signals in accordance with Bluetooth Specification 2.1+EDR. The basic documents for this specification are regulations for Europe, Japan and North America. The frequency band defined for Bluetooth devices is the unlicensed 2.4 GHz ISM (Industrial Scientific medical) frequency band.

Two modulation modes are used for Bluetooth: a mandatory mode, called the Basic Rate and an optional one, called the Enhanced Data Rate. The Basic Rate mode uses binary FM modulation and has data rate of 1 Mbps. The Enhanced Data Rate uses two types of PSK modulation, the $\pi/4$ -DQPSK or 8DPSK, and achieves data rates of 2 Mbps and 3 Mbps, respectively. All modulations schemes have the symbol rate equal to 1Ms/s.

A Time Division Duplex (TDD) scheme for full duplex transmission is defined for both modes.

The latest Bluetooth Low Energy specification is supported as well. See [chapter 3.4, "Bluetooth Low Energy"](#), on page 25 for a detailed description on Bluetooth Low Energy support.

The following list gives an overview of the options provided by the R&S Signal Generator for generating of signals according to the Bluetooth specification:

- Support for three transport modes, the ACL+EDR, SCO, eSCO+EDR transport modes
- Support of all packet types for both the Basic Rate and the Enhanced Data Rate modes
- Generation of signals with up to 5238 frames sequence length
- Configuration of the packet contents with a convenient packet editor or all data packets, both with optional data whitening
- Generation of signals in accordance to the Dirty Transmitter Test specification for both the basic and Enhanced Data Rates and with possibilities to change the start phase, the frequency drift rate and the frequency drift deviation
- Power Ramp Control with possibilities to choose ramp time, rise and fall offset
- Configuration of the clipping, filter and modulation settings

2.1 Bluetooth Transport Modes

There are three different transport modes defined in the Bluetooth specification, each of them with special applications:

- Synchronous Connection-Oriented (SCO)
The SCO transport mode is used for a symmetric point-to-point link establishment between a master and a specific slave in the piconet.
- Extended Synchronous Connection-Oriented (eSCO)
The eSCO transport mode is used for a symmetric or asymmetric, point-to-point link establishment between the master and a specific slave.

- Asynchronous Connection-Less (ACL)
The ACL transport mode is used for a point-to-multipoint link establishment between the master and all slaves participating on the piconet.

There are some common transmitted packets used by all transport modes and some specific packets defined for each transport mode.

2.2 Bluetooth Packet Types

2.2.1 ACL packets

The ACL packets are used for asymmetric links and they contain user data or control data. The table and the figures below give an overview of the ACL packets and their structure.

Table 2-1: ACL packet - Basic Rate

Type	Payload Header (bytes)	User Payload (bytes)	FEC	CRC	Slot Number
DM1	1	0-17	2/3	Yes, 16-bit	1
DH1		0-27	no		
DM3	2	0-121	2/3		3
DH3		0-183	no		
DM5		0-224	2/3		5
DH5		0-339	no		
AUX1	1	0-29		no	



Fig. 2-1: Packet Structure of ACL packets - Basic Rate

Table 2-2: ACL packets - Enhanced Rate

Type	Payload Header (bytes)	User Payload (bytes)	FEC	CRC	Slot Number
2-DH1	2	0-54	no	Yes, 16-bit	1
2-DH3		0-367			3
2-DH5		0-679			5
3-DH1		0-83			1
3-DH3		0-552			3
2-DH5		0-1021			5



Fig. 2-2: Packet Structure of ACL packets - Enhanced Rate

2.2.2 SCO and eSCO packets

The SCO and eSCO packets are used for symmetric links. The SCO packets are used for 64 kb/s speech transmission and for transparent synchronous data. The eSCO packets are also used for 64kb/s speech transmission and transparent data at 64 kb/s but also at other rates.

The tables and the figures below give an overview of the SCO and eSCO packets and their structure.

Table 2-3: SCO packets

Type	Payload Header (bytes)	User Payload (bytes)	FEC	CRC	Slot Num ber
HV1	n.a.	10	1/3	no	n.a.
HV2		20	2/3		
HV3		30			
DV	1 (Data only)	10+(0-9)	2/3 (Data only)	Yes, 16-bit (Data only)	

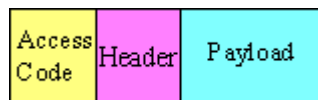


Fig. 2-3: Packet Structure SCO packets

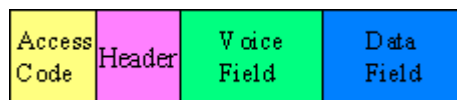


Fig. 2-4: Packet Structure SCO packets (data only)

Table 2-4: eSCO packets - Basic Rate

Type	Payload Header (bytes)	User Payload (bytes)	FEC	CRC	Slot Number
EV3	n.a.	1-30	no	Yes, 16-bit (Data only)	1
EV4		1-120	2/3		3
EV5		1-180	no		3

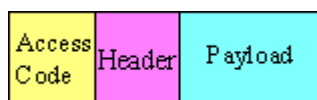


Fig. 2-5: Packet Structure eSCO packets - Basic Rate

Table 2-5: eSCO packets - Basic Rate

Type	Payload Header (bytes)	User Payload (bytes)	FEC	CRC	Slot Number
2-EV3	n.a.	1-60	no	Yes, 16-bit	1
2-EV5		1-360			3
3-EV3		1-90			1
3-EV5		1-540			3

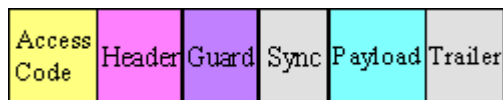


Fig. 2-6: Packet Structure eSCO packets - Enhanced Rate

2.2.3 Link control packets for ACL, SCO, eSCO transport modes

There are some common kinds of packet types. An overview of these packet types is given in the table below.

Table 2-6: Common link control packets

Transport modes	Type	Payload Header (bytes)	FEC	CRC	Application
SCO,eSCO,ACL	ID	n.a.	n.a.	n.a.	Paging, inquiry, response
SCO,eSCO,ACL	NULL				Carries Link information to the source, e.g. about successfully received signal (ARQN) or the state of the receiving buffer (FLOW)
SCO,eSCO,ACL	POLL				Similar to NULL packet, used by master to poll the slaves, must be confirmed
SCO,ACL	FHS	18	2/3	Yes	Page master response, inquiry response, in roll switch

Table 2-7: Common link control packets: packet structure

Packet Type ID	Packet Types NULL and POLL	Packet Types FHS
Access Code (DAK or IAC)	Access Code Header	Access Code Header Payload

2.3 Packet Structure and Fields

Allmost all Bluetooth transmitted packets have standard format and consist of the access code, the header and the payload with useful information. The exceptions are the ID packet which consists of the access code only and NULL and POLL packets which carry only the access code and the header.

2.3.1 Access code

The access code is used for synchronisation, DC offset compensation and identification. The fields of the access code are shown in the figure below and their meaning is explained in the table below.

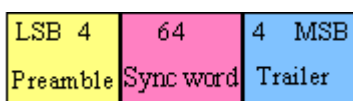


Table 2-8: The access code fields

Field	Description	Packets
Preamble	A fixed zero-one pattern of 4 symbols, used to facilitate DC compensation	All packets
Sync Word	A 64-bit code word derived from a 24 bit address, improves timing acquisition	All packets
Trailer	A fixed zero-one pattern of four symbols, extended DC compensation	All packets, except ID

2.3.2 Header

The Header contains link control information. The fields of the header are shown in the figure and their meaning is explained in the table below.

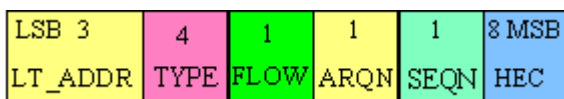


Table 2-9: The header fields

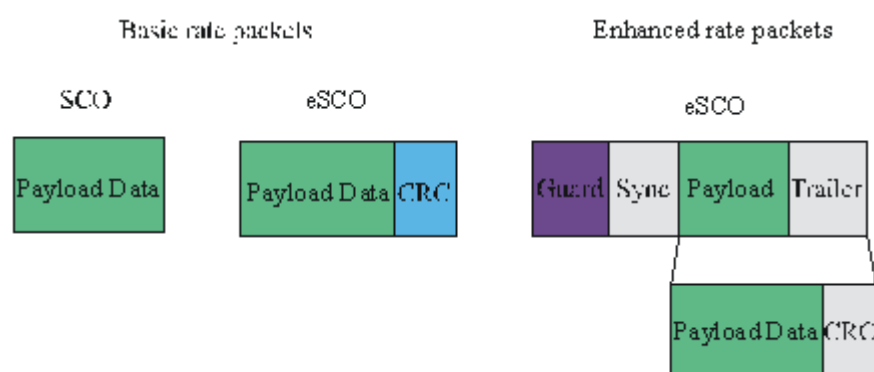
Field	Description	Packets
LT_ADDR	Logical transport address, indicates the destination slave for a packet in a master-to-slave transmission slot and the source slave for a slave-to-master transmission slot	
TYPE	Type code, specifies which packet type is used	
FLOW	Flow control, used for flow control of packets over the ACL logical transport. When the RX buffer in the recipient is full, a STOP indication shall be returned. When the RX buffer can accept data, a Go indication shall be returned.	All packets, except ID
ARQN	Automatic Repeat Request Number, acknowledgement indication, used to inform the source of a successful transfer of payload data with CRC can be positive acknowledged ACK or negative acknowledged NAK,	

Field	Description	Packets
SEQN	Sequential numbering scheme to order the data packet stream	
HEC	Header-error-check to check the header integrity	

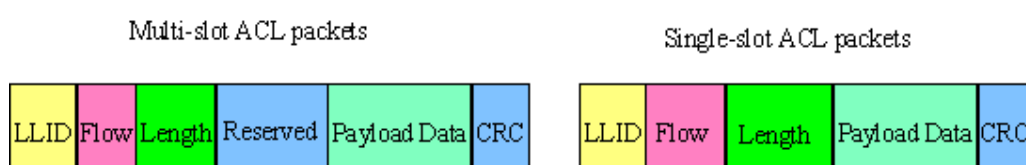
2.3.3 Payload format

The payload structure depends on the type of the data field and the data rate. Two fields are defined in the payload: the synchronous data field and the asynchronous data field. The ACL packets only have the asynchronous data field and the SCO and eSCO packets only have the synchronous data field. The exception is DV of SCO transport mode which has both data fields, synchronous and asynchronous.

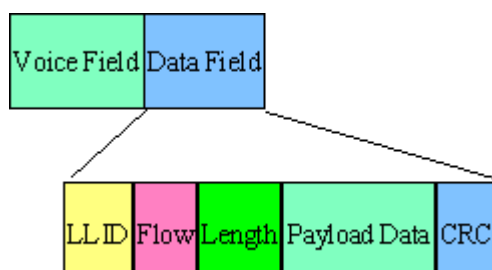
Synchronous data fields



Asynchronous data fields



Synchronous and Asynchronous data fields

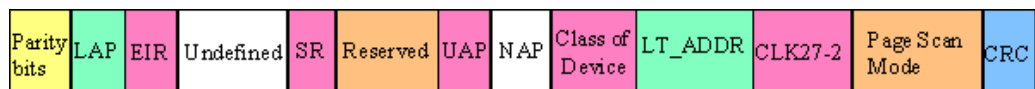


The meaning of some payload fields is given in the table below.

Table 2-10: The payload fields

Field	Description
CRC	The cyclic redundancy error check
Guard, Sync	The guard time and synchronization sequence, used for physical layer change of modulation scheme
LLID	The logical link identifier, specifies the logical link
Flow	Field which controls the flow on the logical channels

The payload format and content of the FHS packet are different from other packets. The fields of the FHS packet are shown in the figure below and their meaning is explained in the table below.

**Table 2-11: The payload fields for the FHS packet**

Field	Description
Parity bits	Form the first part of the sync word of the access code of the device that sends the FHS packet
LAP	Contains the lower address part of the device that sends the FHS packet
EIR	An extended inquiry response, provides miscellaneous information during the inquiry response procedure
Undefined	Reserved for future use and shall be set to zero
SR	The scan repetition field, indicates the interval between two consecutive page scan windows
Reserved	Shall be set to 10
UAP	Contains the upper address part of the device that sends the FHS packet
NAP	Contains the non-significant address part of the device that sends the FHS packet
Class of device	Contains the class of device of the device that sends the FHS packet. This field is defined in Bluetooth Assigned Numbers.
LT_ADDR	Contains the logical transport address
CLK27-2	Contains the value of the native clock of the device that sends the FHS packet, sampled at the beginning of the transmission of the access code of this FHS packet
Page scan mode	Indicates which scan mode is used by default by the sender of the FHS packet

2.4 Bluetooth Modulation Schemes

The modulation used for the basic data rate packets is GFSK (Gaussian Frequency Shift Keying) with a bandwidth bit period product $BT=0.5$. The modulation index is between 0.28 and 0.35.

The modulation scheme used for enhanced data rate packets changes within the packet. The access code and packet header has GFSK modulation scheme and are transmitted with the Basic Rate 1Mbps, while the subsequent synchronisation sequence, payload and trailer sequence have a PSK type of modulation and are transmitted with a data rate of 2 Mbps or optionally 3 Mbps.

The PSK modulation, namely $\pi/4$ rotated differential encoded quaternary phase shift keying ($\pi/4$ -DQPSK) is defined for the 2 Mbps transmission.

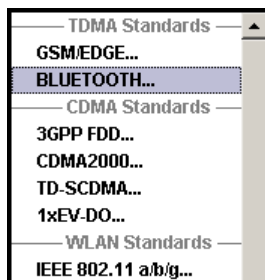
The PSK modulation, namely differential encoded 8-ary phase shift keying (8DPSK), is defined for the 3Mbps transmission.

The modulation types and corresponding packet types are given in the table below.

Table 2-12: The modulation types and corresponding packet types

Modulation type	Packet types
GFSK	ID, NULL, POLL, FHS, DM1, DH1, DM3, DH3, DM5, DH5, AUX1, HV1, HV2, HV3, DV, EV3, EV4, EV5
GFSK + $\pi/4$ -DQPSK	2-DH1, 2-DH3, 2-DH5, 2-EV3, 2-EV5
GFSK + 8DPSK	3-DH1, 3-DH3, 3-DH5, 3-EV3, 3-EV5

3 User Interface



The menu for setting the Bluetooth digital standard is either called from the baseband block or from the menu tree under "Baseband".

The menu is split into several sections for configuring the standard.

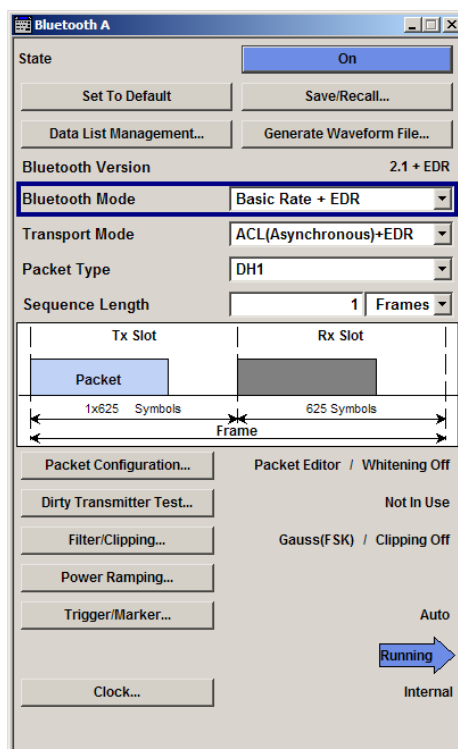
The upper section of the menu is where the Bluetooth digital standard is enabled, the default settings are called, and the transport mode, the packet type and the sequence length are selected.

The valid Bluetooth version in use is displayed.

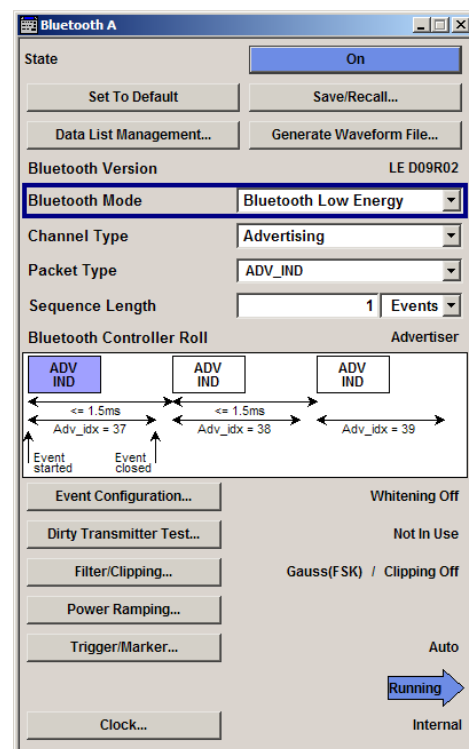
Between an upper part of menu and a lower part of menu Transmission/ ReceptionTX/RX timing is shown. Transmitted packet has duration $N \times 625 \mu s$ where N is an odd integer larger than 0. N depends on the type of the transmitted packet. For Bluetooth Low Energy modes, various packet durations apply depending on the type of channel and packet selected.

Many of the buttons lead to submenus for loading and saving the Bluetooth configuration and for setting the filter, trigger, and clock parameters.

Specific settings of the Bluetooth modes are described in separate chapters. Refer to [chapter 3.2, "Bluetooth Basic Rate + EDR"](#), on page 20 and [chapter 3.4, "Bluetooth Low Energy"](#), on page 25.



Bluetooth Basic Rate + EDR



Bluetooth Low Energy



The screenshots provided in this description show parameter values that have been selected to illustrate as much as possible of the provided functions and possible inter-dependencies between them.

These values are not necessarily representative of realistic test situations.

3.1 General Settings for Bluetooth Signals

The upper menu section is where the Bluetooth digital standard is enabled and reset and where the transport mode, packet type and sequence length are selected.

State

Activates or deactivates the Bluetooth standard.

Enabling this standard disables all the other digital standards and digital modulation modes.

In case of two-path instruments, this affects the same path.

Remote command:

[\[:SOURce<hw>\]:BB:BT0oth:STATe](#) on page 68

Set To Default

Calls the default settings.

Parameter	Value
Bluetooth Version	2.1 + EDR (2.1 = current version number)
Bluetooth Mode	Basic Rate + EDR
Transport mode	ACL (Asynchronous) + EDR
Packet type	DH1
Sequence length	1 Frames
Packet configuration	Packet Editor/ Whitening off
Dirty Transmitter Test	Not in Use
Filter/Clipping	Gauss (FSK) / Clipping off
Power ramping	Cosine / 1 Symbols
Trigger/Marker	Auto
Clock	Internal

Remote command:

[\[:SOURce<hw>\]:BB:BT0oth:PRESet](#) on page 65

Save/Recall...

Calls the "Save/Recall" menu.

From the "Save/Recall" menu, the "File Select" windows for saving and recalling Bluetooth configurations and the "File Manager" are called.



Bluetooth configurations are stored as files with the predefined file extension *.bto. The file name and the directory they are stored in are user-definable.

The complete settings in the Bluetooth menu are saved and recalled.

"Recall Bluetooth Setting"	Opens the "File Select" window for loading a saved Bluetooth configuration. The configuration of the selected (highlighted) file is loaded by pressing the "Select" button.
"Save Bluetooth Setting"	Opens the "File Select" window for saving the current Bluetooth signal configuration. The name of the file is specified in the "File name" entry field, the directory selected in the "Save Into" field. The file is saved by pressing the "Save" button. The "Fast Save" checkbox determines whether the instrument performs an absolute or a differential storing of the settings. Enable this function to accelerate the saving process by saving only the settings with values different to the default ones. "Fast Save" is not affected by the "Preset" function.
"File Manager"	Calls the "File Manager". The "File Manager" is used to copy, delete, and rename files and to create new directories.

Remote command:

[:SOURCE<hw>] :BB:BT00th:SETTing:CATalog on page 66
 [:SOURCE<hw>] :BB:BT00th:SETTing:LOAD on page 67
 [:SOURCE<hw>] :BB:BT00th:SETTing:STORe on page 67
 [:SOURCE<hw>] :BB:BT00th:SETTing:STORe:FAST on page 67
 [:SOURCE<hw>] :BB:BT00th:SETTing:DELeTe on page 66

Data List Management...

Calls the "Data List Management" menu. This menu is used to create and edit a data list.



All data lists are stored as files with the predefined file extension *.dm_iqd. The file name and the directory they are stored in are user-definable.

The data lists must be selected as a data source from the submenus under the individual function, e.g. in the channel table of the cells.

Note: All data lists are generated and edited by means of the `SOURCE:BB:DM` subsystem commands. Files containing data lists usually end with `*.dm_iqd`. The data lists are selected as a data source for a specific function in the individual subsystems of the digital standard.

Example: Creating and editing the data list

```
SOUR:BB:DM:DLIS:SEL 'd_list1'
SOUR:BB:DM:DLIS:DATA #B1111010101000001111....
SOUR:BB:DM:DLIS:DATA:APP #B1111010101000001111....
```

Remote command:

[\[:SOURCE<hw>\]:BB:BT0oth:PCONfiguration:DATA](#) on page 92

[\[:SOURCE<hw>\]:BB:BT0oth:PCONfiguration:DATA:DSElection](#) on page 93

Generate Waveform File...

Calls the "Generate Waveform" menu. This menu is used to store the current Bluetooth signal as ARB signal in a waveform file.

This file can be loaded in the "ARB" menu and processed as multicarrier or multisegment signal.

The file name is entered in the submenu. The file is stored with the predefined file extension `*.wv`. The file name and the directory it is stored in are user-definable.

Remote command:

[\[:SOURCE<hw>\]:BB:BT0oth:WAVEform:CREate](#) on page 69

Bluetooth Version

Displays the current version of the Bluetooth / Bluetooth LE standard.

The default settings and parameters provided are oriented towards the specifications of the version displayed.

Remote command:

[\[:SOURCE<hw>\]:BB:BT0oth:VERSion](#) on page 69

Bluetooth Mode

Determines the digital Bluetooth standard. Basic Rate + EDR or Bluetooth Low Energy are available.

"Basic Rate +EDR"

Selects the Bluetooth mode Basic Rate + EDR.

Specific settings of the basic mode are described in [chapter 3.2, "Bluetooth Basic Rate + EDR"](#), on page 20.

"Bluetooth Low Energy"

Selects the Bluetooth mode Low Energy. The settings concerning Bluetooth Low Energy mode are described in [chapter 3.4, "Bluetooth Low Energy"](#), on page 25.

Remote command:

[\[:SOURCE<hw>\]:BB:BT0oth:BMODE](#) on page 106

Filter/Clipping

Calls the dialog for setting baseband filter and clipping of the generated signal. The current filter and the clipping state are displayed next to the button.

The dialog is described in [chapter 3.8, "Filter/Clipping Settings"](#), on page 48.

Remote command:

n.a.

Power Ramping

Calls the dialog for setting the power ramping.

The dialog is described in [chapter 3.9, "Power Ramping Settings"](#), on page 52.

Remote command:

n.a.

Trigger/Marker

(Trigger for R&S SMx/AMU instruments only)

Calls the dialog for selecting the trigger mode and trigger source, for configuring the marker signals, and for setting the time delay of an external trigger signal.

The currently selected trigger mode and trigger source are displayed next to the button.

The dialog is described in [chapter 3.10, "Trigger/Marker/Clock Settings"](#), on page 53.

Remote command:

n.a.

Execute Trigger

Executes a trigger manually. A manual trigger can be executed only if an internal trigger source and a trigger mode other than "Auto" have been selected.

Remote command:

[\[:SOURce<hw>\]:BB:BT0oth:TRIGger:EXECute](#) on page 77

Arm

Stops signal generation manually. This button appears only with "Running" signal generation in the "Armed_Auto" and "Armed_Retrigger" trigger modes.

Signal generation can be restarted by a new trigger (internally with "Execute Trigger" or externally).

Remote command:

[\[:SOURce<hw>\]:BB:BT0oth:TRIGger:ARM:EXECute](#) on page 76

Clock

Calls the dialog for selecting the clock source and for setting a delay, see [chapter 3.10, "Trigger/Marker/Clock Settings"](#), on page 53.

Remote command:

n.a.

3.2 Bluetooth Basic Rate + EDR

The dialog comprises the parameters to define the packet type and access the packet type configuration dialog. The graphic shows the frame structure of the selected packet type.

Transport Mode

Only available for "Bluetooth Mode " set to "Basic Rate + EDR"

Selects the transport mode.

- | | |
|------------|--|
| "ACL+EDR" | The transport mode selected is used for a point-to-multipoint link establishment between the master and all the slaves participating on the piconet. |
| "SCO" | The transport mode selected is used for a point-to-point link establishment between a master and a single slave in the piconet. |
| "eSCO+EDR" | The transport mode selected is used for a symmetric or asymmetric point-to-point link establishment between a master and a specific slave. |

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:TM0De` on page 68

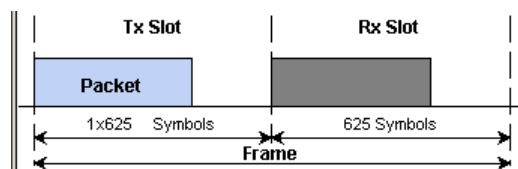
Packet Type

Selects the packet type.

The available packets depend on the selected [Transport Mode](#).

All packet types as defined in the Bluetooth specification are supported. For an overview, see [chapter 2.2, "Bluetooth Packet Types"](#), on page 8.

The graphic shows the frame structure of the selected packet type.



The transmitted packet has a duration of $N \times 625 \mu s$ where N is an odd integer larger than 0. N depends on the type of the transmitted packet.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:PTYPE` on page 66

Sequence Length

Selects the sequence length in frames of the generated signal. The signal repeats after the specified number of frames.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:SENGth` on page 68

Packet Configuration

Access the "Packet Configuration" dialog, see [chapter 3.3, "Packet Configuration - Bluetooth Basic Rate + EDR"](#), on page 21.

The current data source for packet and the data whitening state are displayed next to the button.

Remote command:
n.a.

3.3 Packet Configuration - Bluetooth Basic Rate + EDR

The "Packet Configuration" dialog is reached via the "Bluetooth" main dialog.

The available settings in the lower part of the dialog depend on the selected data source for packet and the selected packet type. Header configurations are selected in the "Header" settings section.

Provided are the following settings:

Data Source for Packet

The data sent for each packet can be comfortably edited with the Packet Editor, or filled with a predefined ALL Data sequence.

"Packet Editor" Enables the edit mode to configure the packet fields individually.

"All Data" Fills the generated packets with the selected data source. This mode is useful if you need to load predefined data contents from a data list file or the data contents of the packet are not of interest.

Remote command:

[:SOURce<hw>] :BB:BT00th:PCONfiguration:DSFPacket on page 94

Data Whitening

Activates or deactivates the Data Whitening.

Evenly distributed white noise is ideal for the transmission and real data can be forced to look similar to white noise with different methods called Data Whitening.

Remote command:

[\[:SOURCE<hw>\]:BB:BT00th:PCONfiguration:DWhitening](#) on page 95

Bluetooth Device Address (BD_ADDR)

Enters Bluetooth Device Address. Each Bluetooth device shall be allocated a unique 48-bit Bluetooth device address (BD_ADDR).

ACDE	48	0000 80
NAP	UAP	LAP

The BD_ADDR may take any values except the 64 reserved LAP values: 0x9E8B00 – 0x9E8B3F.

"NAP" Selects non-significant address part.
The length of NAP is 16 bits or 4 hexadecimal figures.

"UAP" Selects upper address part.
The length of UAP is 8 bits or 2 hexadecimal figures.

"LAP" Selects lower address part.
The length of LAP is 24 bits or 6 hexadecimal figures.

Remote command:

[\[:SOURCE<hw>\]:BB:BT00th:PCONfiguration:BDANap](#) on page 91

[\[:SOURCE<hw>\]:BB:BT00th:PCONfiguration:BDAUap](#) on page 92

[\[:SOURCE<hw>\]:BB:BT00th:PCONfiguration:BDALap](#) on page 91

Logical Transport Address

(Available for all packet types except ID)

Enters the logical transport address for the header.

Each slave active in a piconet is assigned a primary logical transport address (LT_ADDR). The all-zero LT_ADDR is reserved for broadcast messages.

Remote command:

[\[:SOURCE<hw>\]:BB:BT00th:PCONfiguration:LTAddress](#) on page 96

Flow Control (Header)

(Available for all packet types except ID)

Sets the FLOW bit in the header. This bit indicates start or stop of transmission of packets over the ACL logical transport.

"Go" Allows the other devices to transmit new data.

"Stop" Stops the other devices from transmitting data temporarily.

Remote command:

[\[:SOURCE<hw>\]:BB:BT00th:PCONfiguration:HFControl](#) on page 96

Acknowledgment

(Available for all packet types except ID)

Sets the ARQN bit of the packet header.

"NAK" Request to retransmit the previous payload.

"ACK" Previous payload has been received successfully.

Remote command:

[\[:SOURCE<hw>\]:BB:BT0oth:PCONfiguration:ACKNowledgement](#) on page 91

SEQN Start Value

(Available for all packet types except ID)

Sets the start value of the header SEQN bit.

The SEQN bit is present in the header to filter out retransmissions in the destination. The signal generator is altering this bit automatically on consecutive frames, if a sequence length of at least 2 frames is set.

"0" The SEQN bit starts with 0.

"1" The SEQN bit starts with 1.

Remote command:

[\[:SOURCE<hw>\]:BB:BT0oth:PCONfiguration:SNValue](#) on page 97

Data Source

(Available for all packet types except ID, POLL, NULL and FHS packets)

Selects the data source used for the payload.

The following standard data sources are available:

- "All 0, All 1"
An internally generated sequence containing 0 data or 1 data.
- "PNxx"
An internally generated pseudo-random noise sequence.
- "Pattern"
An internally generated sequence according to a bit pattern.
Use the "Pattern" box to define the bit pattern.
- "Data List/Select DList"
A binary data from a data list, internally or externally generated.
Select "Select DList" to access the standard "Select List" dialog.
 - Select the "Select Data List > navigate to the list file *.dm_iqd > Select" to select an existing data list.
 - Use the "New" and "Edit" functions to create internally new data list or to edit an existing one.
 - Use the standard "File Manager" function to transfer external data lists to the instrument.

See also "Main Dialog > Data List Management".

Remote command:

[\[:SOURCE<hw>\]:BB:BT0oth:PCONfiguration:DATA](#) on page 92

[\[:SOURCE<hw>\]:BB:BT0oth:PCONfiguration:DATA:DPATtern](#) on page 93

[\[:SOURCE<hw>\]:BB:BT0oth:PCONfiguration:DATA:DSElection](#) on page 93

Data Length

(Available all packet types except ID, POLL, NULL and FHS packets)

Enters the payload data length in bytes.

Remote command:

`[ :SOURCE<hw> ] :BB:BT00th:PCONfiguration:DLENgth` on page 94

Flow Control (Payload)

(Available for all packets types except ID, POLL, NULL, FHS, HV1, HV2, HV3, EV3, EV4, EV5, 2-EV3, 2-EV5, 3-EV3, 3-EV5 packets.)

Sets the FLOW bit in the payload (flow control per logical link)

"Go" Indicates start of transmission of ACL packets after a new connection has been established.

"Stop" Indicates stop of transmission of ACL packets before an additional amount of payload data is sent.

Remote command:

`[ :SOURCE<hw> ] :BB:BT00th:PCONfiguration:PFControl` on page 96

Packet Length

(Available in All Data mode only and for all packet types except ID packet)

Enters the packet length in symbols.

Remote command:

`[ :SOURCE<hw> ] :BB:BT00th:PCONfiguration:PLENgth` on page 97

EIR packet follows

(Available for FHS packets)

Indicates that an extended inquiry response packet may follow.

"Yes" Indicates that an EIR packet follows.

"No" Indicates that EIR does not follow.

Remote command:

`[ :SOURCE<hw> ] :BB:BT00th:PCONfiguration:EIRPacketfollows`
on page 95

Scan Repetition Mode

(Available for FHS packets)

The 2-bit scan repetition field indicates the interval between two consecutive page scan windows, determines the behavior of the paging device.

"R0" The scan interval is equal to the scan window T_w page scan (continuous scan) and maximal 1.28s.

"R1" The scan interval is maximal 1.28s.

"R2" The scan interval is maximal 2.56s.

Remote command:

`[ :SOURCE<hw> ] :BB:BT00th:PCONfiguration:SRMode` on page 98

Class of Device

(Available for FHS packets)

A parameter received during the device discovery procedure, indicates the type of device and which types of service that are supported.

Remote command:

[\[:SOURCE<hw>\]:BB:BT0oth:PCONfiguration:CODevice](#) on page 92

Data Source (Voice Field)

(Available for DV packets)

Selects the Data Source for the voice field.

The following standard data sources are available:

- "All 0, All 1"
An internally generated sequence containing 0 data or 1 data.
- "PNxx"
An internally generated pseudo-random noise sequence.
- "Pattern"
An internally generated sequence according to a bit pattern.
Use the "Pattern" box to define the bit pattern.
- "Data List/Select DList"
A binary data from a data list, internally or externally generated.
Select "Select DList" to access the standard "Select List" dialog.
 - Select the "Select Data List > navigate to the list file *.dm_iqd > Select" to select an existing data list.
 - Use the "New" and "Edit" functions to create internally new data list or to edit an existing one.
 - Use the standard "File Manager" function to transfer external data lists to the instrument.

See also "Main Dialog > Data List Management".

Remote command:

[\[:SOURCE<hw>\]:BB:BT0oth:PCONfiguration:VDATa](#) on page 98

[\[:SOURCE<hw>\]:BB:BT0oth:PCONfiguration:DATA:VDPattern](#) on page 93

[\[:SOURCE<hw>\]:BB:BT0oth:PCONfiguration:DATA:VDSElection](#)

on page 94

3.4 Bluetooth Low Energy

The R&S Signal Generator provides you with ability to generate signals in accordance with Bluetooth Low Energy Specification (LE). Bluetooth Low Energy makes it possible to transfer data from low power devices running on the smallest of batteries to a larger device, such as a PC, a mobile phone, or a PDA. For the first time, a Bluetooth connection to a wristwatch, or a heart rate sensor, or a data transfer from a digital camera, is possible. The Bluetooth low energy chips will offer capabilities that do not replace or supersede the existing Bluetooth 2.x standards. Data rates are comparable to Bluetooth 1.1, and are data-only (no audio content).

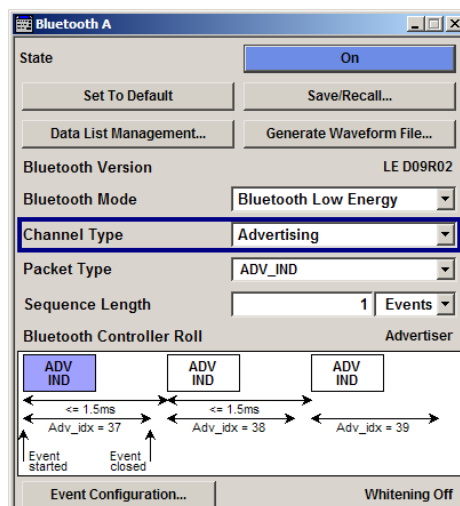
Bluetooth LE mode only uses the Basic Rate. The Basic Rate mode uses binary FM modulation and has a data rate of 1 MBps. The modulation scheme has the symbol rate equal to 1Ms/s.

For full duplex transmission, a Time Division Duplex (TDD) scheme is used.

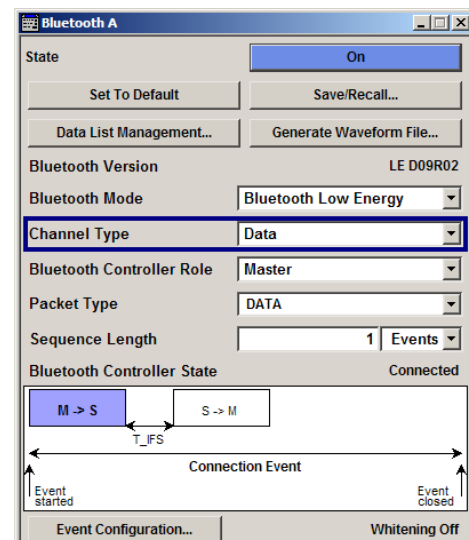
The following list gives an overview of the options provided by the R&S Signal Generator for Bluetooth LE compliant signal generation:

- Support for two channel types, the Advertising and Data channel types.
- Support of all Bluetooth LE packet types.
- Sequence Length unit can be set to Event or Frame.
- Convenient packet editor for all supported packet types including optional data whitening.
- Dirty Transmitter Test compliant to RF test specification, with options to change start phase, frequency drift rate and frequency drift deviation.
- Power Ramp Control with configurable ramp time, rise and fall offsets.
- Clipping, filter and modulation settings supported.

In the following description Bluetooth Low Energy is abbreviated as Bluetooth LE.



Bluetooth LE Channel Type Advertising



Bluetooth LE Channel Type Data

Channel Type

Determines the channel type. Advertising and data are available. Refer to [chapter 3.5, "Event / Frame Configuration - Bluetooth LE"](#), on page 29 for setting the respective parameters.

"Advertising" Selects channel type Advertising.

"Data" Selects channel type Data.
Devices in a connected state transmit data channel packets in connection events with a start point and an interval.

Remote command:

[:SOURce<hw>] :BB:BT00th:CTYPE on page 65

Packet Type

Selects the packet type.

The available packet types depend on the selected channel type, as shown in the table below ([table 3-1](#)).

Table 3-1: Packet types of the respective channel types:

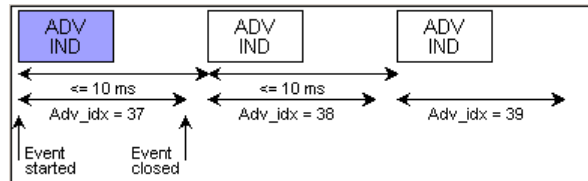
Packet Type	Advertising	Data
ADV_IND/	x	-
ADV_DIRECT_IND	x	-
ADV_NONCONN_IND	x	-
ADV_DISCOVER_IND	x	-
SCAN_REQ	x	-
SCAN_RSP	x	-
CONNECT_REQ	x	-
DATA	-	x
CONTROL_DATA ...	-	x
TEST PACKET	x	x

Depending on the [Bluetooth Controller Role](#) (master or slave), you can determine in detail the information of the "CONTROL_DATA", as shown in the following table [table 3-2](#).

Table 3-2: Control information, available for master or slave.

CONTROL_DATA	Master	Slave
LL_CONNECTION_UPDATE_REQ	x	-
LL_CHANNEL_MAP_REQ	x	-
LL_TERMINATE_IND	x	-
LL_ENC_REQ	x	-
LL_RNC_RESP	-	x
LL_START_ENC_REQ	x	x
LL_START_ENC_RESP	x	x
LL_UNKNOWN_RESP	-	x
LL_FEATURE_REQ	x	-
LL_FEATURE_RESP	-	x
LL_PAUSE_ENC_REQ	x	-
LL_PAUSE_ENC_RESP	-	x
LL_VERSION_IND	x	x
LL_REJECT_IND	x	x

The graphic shows the frame structure of the selected packet type.



Remote command:

`[ :SOURCE<hw> ] :BB:BT00th:UPTYPE` on page 124

Sequence Length

Selects the number of frames or events depending on the packet type. The signal repeats after the specified number of frames/events.

For SCAN_REQ and CONNECT_REQ packet, the sequence length is expressed in "Frames".

For TERMINATE_IND packets, a default value according to the specification is given:

- Master: 'SlaveLatency + 6'
- Slave: '6'

For all other packet types the sequence length is expressed in "Events".

Remote command:

`[ :SOURCE<hw> ] :BB:BT00th:USLength` on page 125

Bluetooth Controller Role

Determines the controller role.

Note: The available packet types for the selected channel types and the controller roles are described in "[Bluetooth Controller Role](#)" on page 28.

Depending on the channel type, the field either displays the appropriate role or you can select one:

- "Advertiser"
 - Displays the controller role corresponding to the packet type:
 - "Advertiser" for all ADV-xxx packet types and SCAN_RSP
 - "Scanner" for SCAN_REQ packet type
 - "Initiator" for CONNECT_REQ packet type
- "Data"
 - Assigns a role to the controller:
 - "Master"
 - "Slave"

Remote command:

`[ :SOURCE<hw> ] :BB:BT00th:BCRole` on page 105

Bluetooth Controller State

Shows the state of the bluetooth controller for channel type "Data".

Remote command:

`[ :SOURCE<hw> ] :BB:BT00th:BCText?` on page 64

Event / Frame Configuration

Calls a submenu for setting Event Configuration, depending on the packet type. In case the sequence length of the packet type is expressed in frames, the button calls the menu for setting the Frame Configuration

These submenus are described in [chapter 3.5, "Event / Frame Configuration - Bluetooth LE"](#), on page 29.

The data whitening state is displayed next to the button.

Remote command:

n.a.

Test Packet Configuration

Access the "Test Packet Configuration" dialog for packet type "TEST PACKET", see [chapter 3.6.2, "Test Packet Configuration Settings"](#), on page 44.

3.5 Event / Frame Configuration - Bluetooth LE

The "Event/ Frame Configuration" dialogs are reached via the "Bluetooth" main dialog.

The upper area of the dialogs changes, as a function of the used channel type. For channel type "Data", the section provides "Connection Settings", and for channel type "Advertising" the "Advertising Settings" are given with regard to this channel type. The "Packet Configuration" subdialog can be opened, and a graph represents the physical channel mapping and the channel indices. The table in the lower section gives an overview of the used channels and their parameters.

Bluetooth A: Event Configuration

Connection Settings

No. of Tx Packets per Event: Connection Event Interval: ms

LL Connection Mode: Long Term Key(hex):

Selected Data Channel Index:

Packet Configuration...

Center frequency(MHz)

2402 2404 2424 2426 2428 2478 2480

ADV Data Data ADV Data Data ADV

37 0 10 38 11 36 39

Channel Index

Channel Table

	Center Frequency[MHz]	Channel Index	Channel Type	Channel State
Channel 0	2 404	0	Data	Used
Channel 1	2 406	1	Data	Unused
Channel 2	2 408	2	Data	Unused
Channel 3	2 410	3	Data	Unused
Channel 4	2 412	4	Data	Unused
Channel 5	2 414	5	Data	Unused

Fig. 3-1: Event Configuration Dialog of channel type Data

Bluetooth A: Event Configuration

Advertising Settings

Advertising Event Interval: ms Advertising Event Delay: ms

Packet Configuration...

Center frequency(MHz)

2402 2404 2424 2426 2428 2478 2480

ADV Data Data ADV Data Data ADV

37 0 10 38 11 36 39

Channel Index

Channel Table

	Center Frequency[MHz]	Channel Index	Channel Type	Channel State
Channel 0	2 402	37	Advertising	Used
Channel 1	2 426	38	Advertising	Unused
Channel 2	2 480	39	Advertising	Unused

Fig. 3-2: Event Configuration Dialog of channel type Advertising

3.5.1 Advertising Event / Frame Configuration Settings

Advertising Event Interval

Sets the time interval between two consecutive advertising events, with regard to the starting points.

Note: This parameter is relevant for advertising event configuration and for the packet types ADV-IND, ADV_DIRECT_IND, ADV_NONCONN_IND and ADV_DISCOVER_IND.

Remote command:

[\[:SOURCE<hw>\]:BB:BT0oth:ECONfiguration:AEINterval](#) on page 109

[\[:SOURCE<hw>\]:BB:BT0oth:ECONfiguration:ADINterval](#) on page 108

Advertising Event Delay

Sets a time delay between the start times of two consecutive advertising events. The value is added to the advertising event interval.

Note: This parameter is relevant for advertising event configuration and for the packet types ADV-IND, ADV_NONCONN_IND and ADV_DISCOVER_IND.

Remote command:

[\[:SOURCE<hw>\]:BB:BT0oth:ECONfiguration:AEDelay](#) on page 108

Scan Window

Sets the length of the window during which the scanner is operating in the advertising channel.

Note that the scan window is less or equal to the value of the scan interval.

Note: This parameter is relevant for advertising frame configuration and for the packet type SCAN_REQ.

Remote command:

[\[:SOURCE<hw>\]:BB:BT0oth:ECONfiguration:SWINdow](#) on page 123

Scan Interval

Sets the time interval between the starting points of two consecutive windows during which the scanner is operating in an advertising channel.

Note: This parameter is relevant for advertising frame configuration and for the packet type SCAN_REQ.

Remote command:

[\[:SOURCE<hw>\]:BB:BT0oth:ECONfiguration:SINterval](#) on page 123

Advertising Packet Interval

Sets the time interval between packets starting points of two consecutive packets in the advertising channel.

Note: This parameter is relevant for advertising frame configuration and for the packet type SCAN_RSP.

Remote command:

[\[:SOURCE<hw>\]:BB:BT0oth:ECONfiguration:APINterval](#) on page 109

Transmit Window Offset

Displays the start point of the transmit window.

Note: This parameter is relevant for advertising frame configuration and for the packet type CONNECT_REQ.

This parameter is set in the Packet Configuration, see ["Transmit Window Offset"](#) on page 42.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:WOINfo?` on page 123

Transmit Window Size

Indicates the size of the transmit window, regarding to the start point.

Note that the scan window size is less or equal to the value of the connection interval.

Note: This parameter is relevant for advertising frame configuration and for the packet type CONNECT_REQ.

The parameter is set in the "Packet Configuration" dialog, see ["Transmit Window Size"](#) on page 42.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:WSINfo?` on page 124

Packet Configuration

Opens the dialog for setting the corresponding packet configuration.

This dialog is described in [chapter 3.6.1, "Packet Configuration Settings"](#), on page 36.

Remote command:

n.a.

3.5.2 Data Event Connection Settings

No. of Tx Packets per Event

Sets the number of Tx packets per event. Each connection contains at least one data channel packet. The maximum number of packets per event is determined by the duration of the connection event interval.

Note: This parameter is relevant for data event connection settings.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PNUMber` on page 122

Connection Event Interval

Set the time interval between the start points of two consecutive connection events.

Subsequent transmissions within an event are separated by this parameter in order to separate connecting event starting points in time.

Note: This parameter is relevant for data event connection settings and advertising frame configuration with the packet types CONNECTION_UPDATE_REQ and CONNECT_REQ.

Remote command:

[:SOURCE<hw>] :BB:BT0oth:ECONfiguration:PCONfiguration:CINterval
on page 112

[:SOURCE<hw>] :BB:BT0oth:ECONfiguration:PCONfiguration:NCINterval
on page 117

[:SOURCE<hw>] :BB:BT0oth:ECONfiguration:PCONfiguration:CINterval
on page 112

LL Connection Mode

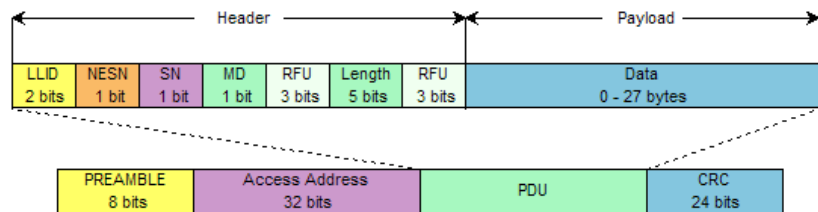
Select the link layer connection mode. In order to provide safe transmission of payload data, the data in the packet can be encrypted. If activated, the payload data follows MIC (Message authentication Code).

Note: This parameter is relevant for data event connection settings.

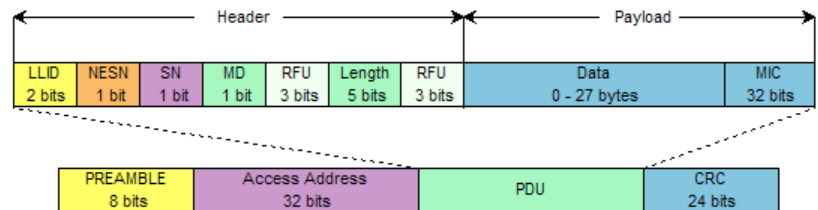
The following table shows which types of packets can be encrypted and / or unencrypted.

Packet Type	encrypted	unencrypted
DATA	X	X
CONNECTION_UPDATE_REQ	X	X
CHANNEL_MAP_REQ	X	X
LL_TERMINATE_IND	X	X
LL_ENC_REQ	-	X
LL_ENC_RSP	-	X
LL_START_ENC_REQ	-	X
LL_START_ENC_RSP	X	-
LL_FEATURE_REQ	X	X
LL_FEATURE_RSP	X	X
LL_PAUSE_ENC_REQ	-	X
LL_PAUSE_ENC_RSP	X	X
LL_VERSION_IND	X	X
LL_REJECT_IND	X	X
UNKNOWN_RSP	X	X

"Unencrypted" Payload data is transmitted without encoding.



"Encrypted" The link layer connection runs in encrypted mode.



Remote command:

`[ :SOURCE<hw> ] :BB:BT00th:ECONfiguration:LCMode` on page 109

Long Term key (hex)

Indicates the time the controller needs to receive the long term key from the host. After this time, the controller is ready to enter into the last phase of encryption mode setup.

Note: This parameter is relevant for data event connection settings. In encrypted mode, the code can be edited.

Remote command:

`[ :SOURCE<hw> ] :BB:BT00th:ECONfiguration:LTKey` on page 110

Selected Data Channel Index

Indicates the number of the first active data channel.

The data channel is selected for each connection event. The master and slave determine the used data channel by selecting from the list of used channels (see "[Channel Table](#)" on page 34).

Note: This parameter is relevant for data event connection settings.

Displays the data channel index currently selected.

Remote command:

n.a.

3.5.3 Channel Table Settings

The channel table displays all parameters characterizing the channel and the current state.

Channel Table

The channel table displays all parameters characterizing the channel and the current state.

Every channel is represented with bit positioned as per the data channel index. LSB represents data channel index 0 and the bit in position 36 represents data channel index 36.

If the channel is used channel its bit is to be set to '1'. Bit value '0' indicates that the channel is unused.

The bits in positions 37, 38 and 39 shall be set to zero upon transmission and ignored upon receipt.

"Center Frequency"

Indicates the center frequency of a channel.

"Channel Index"

Indicates the channel index.

"Channel Type"

Indicates the channel type.

"Channel State"

Indicates used and unused data channels.

Remote command:

Advertising Channel Table: `[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:ACTable:SET<ch>:STATE` on page 108

Data Channel Table: `[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:DCTable:SET<ch>:STATE` on page 108

Data Channel Mapping Table: `[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:DCMTable:SET<ch>:STATE` on page 108

Remote command:

Entire data set for Advertising Channel Table: `[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:ACTable` on page 107

Entire data set for Data Channel Table: `[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:DCTable` on page 107

Entire data set for Channel Map Table: `[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:DCMTable` on page 107

3.6 Packet Configuration - Bluetooth LE

The "Packet Configuration" dialog is reached via the "Event Configuration" dialog or the "Frame Configuration" dialog, respectively.

The available settings in the dialog depend on the selected channel type and the packet type. Header configurations are selected in the "Header" section. Data and control parameters are set in the "Payload" section. The figure shows the packet structure of the currently selected configuration.

3.6.1 Packet Configuration Settings

Data Whitening

Activates or deactivates the Data Whitening.

Evenly distributed white noise is ideal for the transmission and real data can be forced to look similar to white noise with different methods called Data Whitening. Applied to the PDU and CRC fields of all packet types, whitening is used to avoid long equal sequences in the data bit stream.

Remote command:

`[ :SOURCE<hw> ] :BB:BT00th:ECONfiguration:PCONfiguration:DWhitening`
on page 114

Access Address

Sets the access address of the link layer connection.

Bluetooth LE transmissions are based on an interface packet format, that consists of a preamble (8 bits), the access address (32 bits), the PDU and CRC (24 bits).

The access address' structure depends on the packet type:

- Data channel packets
The access address is a pseudo-random LL connection address, generated by the initiator of the LL connection. The address has to follow some specific rules, which are described in the "Bluetooth Low Energy Technology Specification".
- Advertising channel packets
The address is fixed to 0110101101111011001000101110001 with the left most bit sent first and being the LSB.

Note: This parameter is relevant for all available package types specified in events in the data channel, and frames in the advertiser channel, i.e. DATA, CONNECTION_UPDATE_REQ, CHANNEL_MAP_REQ, TERMINATE_IND, LLENC_REQ, LL_FEATURE_REQ, LL_PAUSE_ENC_REQ, LL_START_ENC_RSP, FEATURE_REQ, and CONNECT_REQ.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:AADdress`
on page 110

NESN Start Value

Sets the start value of the next expected packet from the same device in the LL connection (**N**ext**E**xpected **S**equ**E**nce**N**umber). This parameter can be set in the first event. From the second event this field is not indicated.

Note: This parameter is relevant for data event configuration and all data channel packet types except TEST_PACKET.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:NSValue`
on page 118

SN Start Value

Sets the sequence number of the packet. This parameter can be set in the first event. From the second event this field is not indicated.

Note: This parameter is relevant for data event configuration and all data channel packet types except TEST_PACKET.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:SSValue`
on page 120

Device's Addr Type

Selects the address type of the controller device. Depending on the Bluetooth controller role either the Tx or Rx or both address types are assigned.

Subdivided into private and random, a Bluetooth LE device address consists of 48 bits. The format of the device address differs depending on the selected address type.

Note: This parameter is relevant for advertising event or frame configuration.

The bluetooth "Controller Role" and the packet type determine the available entries:

- **Tx** in conjunction with the packet types ADV_IND, ADV_DIRECT_IND, ADV_NON_CONN_IND, ADV_DISCOVER_IND, SCAN_REQ, SCAN_RSP and CONNECT_REQ
- **Rx** for the packet types ADV_DIRECT_IND, SCAN_REQ and CONNECT_REQ

"Public" Allocates a unique 48 bit address to each bluetooth LE device. The public address is given from the registration authority IEEE.

"Random" Allocates a 48 bit address to each bluetooth LE device. A random address is optional.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:TAType`
on page 119

Data Source

Selects the data source used for the payload.

Note: This parameter is relevant for event configuration and packet types DATA, ADV_IND, ADV_NONCONN_IND and ADV_DISCOVER_IND.

The following standard data sources are available:

- "All 0, All 1"
An internally generated sequence containing 0 data or 1 data.
- "PNxx"
An internally generated pseudo-random noise sequence.
- "Pattern"
An internally generated sequence according to a bit pattern.
Use the "Pattern" box to define the bit pattern.
- "Data List/Select DList"
A binary data from a data list, internally or externally generated.
Select "Select DList" to access the standard "Select List" dialog.
 - Select the "Select Data List > navigate to the list file *.dm_iqd > Select" to select an existing data list.
 - Use the "New" and "Edit" functions to create internally new data list or to edit an existing one.
 - Use the standard "File Manager" function to transfer external data lists to the instrument.

See also "Main Dialog > Data List Management".

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:DATA`
on page 112

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:DATA:DPATtern` on page 113

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:DATA:DSElection` on page 113

Data Length

Enters the payload data length in bytes.

Note: This parameter is relevant for event configuration with packet types ADV_IND, ADV_NONCONN_IND and ADV_DISCOVER_IND.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:DLENgth`
on page 114

Connection Event Interval

Set the time interval between the start points of two consecutive connection events. Subsequent transmissions within an event are separated by this parameter in order to separate connecting event starting points in time.

Note: This parameter is relevant for data event connection settings and advertising frame configuration with the packet types CONNECTION_UPDATE_REQ and CONNECT_REQ.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:CINterval`
on page 112

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:NCINterval`
on page 117

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:CINterval`
on page 112

Slave Latency

Sets a number of consecutive connection events the slave can ignore for asymmetric link layer connections.

Note: This parameter is relevant for data event and advertising frame configuration with the packet types CONNECTION_UPDATE_REQ and CONNECT_REQ.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:SLATency`
on page 120

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:NSLatency`
on page 118

LL Connection Timeout

Defines the maximum time between two correctly received Bluetooth LE packets in the LL connection before the connection is considered lost.

Note: This parameter is relevant for data event and advertising frame configuration with the packet types CONNECTION_UPDATE_REQ and CONNECT_REQ.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:LCTimeout`
on page 116

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:NLCTimeout`
on page 117

Connection Instant

Sets a connection instant for indicating the connection event at which the new connection parameters are taken in use.

Both the master and the slave have a 32-bit connection event counter per LL connection. It is reset to zero on the first connection event of the LL connection and incremented by one on every elapsed connection event interval of the LL connection.

Note: This parameter is relevant for data event configuration with the packet types CONNECTION_UPDATE_REQ and CHANNEL_MAP_REQ.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:CINstant`
on page 111

Show / Hide Data Channel (Mapping) Table

In data event and advertising frame configuration with the packet types CHANNEL_MAP_REQ and CONNECT_REQ, calls / hides the channel map table that displays the used channels and their parameters.

The channel table is described in [chapter 3.5.3, "Channel Table Settings"](#), on page 34.

Remote command:

n.a.

Hop Length

Sets the difference from the current channel to the next channel. The master and slave devices determine the data channel in use for every connection event from the channel map. Hop_length is set for the LL connection and communicated in the CONNECT_REQ packets.

Note: This parameter is relevant for data event and advertising frame configuration with the packet type CONNECT_REQ.

Remote command:

`[ :SOURCE<hw> ] :BB:BT00th:ECONfiguration:PCONfiguration:HLength`
on page 115

Random Vector (hex)

Sets the random vector of the master for device identification.

The parameter is an initialization vector provided by the Host in the HCI_ULP_Start_Encryption command.

Note: This parameter is relevant for data event configuration with the packet type LLENC_REQ.

Remote command:

`[ :SOURCE<hw> ] :BB:BT00th:ECONfiguration:PCONfiguration:RVEctor`
on page 119

Encrypted DIVERsifier (hex)

Sets the encrypted diversifier of the master for device identification. The parameter is an initialization vector provided by the Host in the HCI_ULP_Start_Encryption command.

Note: This parameter is relevant for data event configuration with the packet type LLENC_REQ.

Remote command:

`[ :SOURCE<hw> ] :BB:BT00th:ECONfiguration:PCONfiguration:EDIVERsifier`
on page 115

Session Key ID (hex)

Sets the master's or the slave's portion of the session key diversifier (SKDm/SKDsl).

Note: This parameter is relevant for data event configuration with the packet types LLENC_REQ (Master) and LL_ENC_RSP (Slave).

Remote command:

`[ :SOURCE<hw> ] :BB:BT00th:ECONfiguration:PCONfiguration:MSKD`
on page 116

Initialization Vector (hex)

Sets the master's or the slave's portion of the initialization vector(IVm/IVs).

Note: This parameter is relevant for data event configuration with the packet types LLENC_REQ (Master) and LL_ENC_RSP (Slave).

Remote command:

[\[:SOURCE<hw>\]:BB:BT00th:ECONfiguration:PCONfiguration:MIVector](#)
on page 116

Feature Set Length

Enables that the feature set length is indicated.

FeatureSet indicates whether the Controller features are used or not. All the data in FeatureSet is RFU(zero).

Note: This parameter is relevant for data event configuration with the packet types FEATURE_REQ (Master) and FEATURE_RSP (Slave).

Remote command:

[\[:SOURCE<hw>\]:BB:BT00th:ECONfiguration:PCONfiguration:FSLength](#)
on page 115

Unknown Type (hex)

Enables that an invalid control packet is indicated.

The CtrType field indicates the value of the LL control packet that caused the transmission of this packet.

This parameter is relevant for data event configuration with the packet type UNKNOWN_RSP (Slave).

Remote command:

[\[:SOURCE<hw>\]:BB:BT00th:ECONfiguration:PCONfiguration:UTYPE](#)
on page 121

Controller's Device Addr

Sets the advertiser's device address.

In Bluetooth LE systems all the transmissions start with an 8 bit preamble followed by an access address. The access address is composed of a the parts "Company_Id" (LSB) and the "Company_assigned" (MSB). Beside the address fields the notation is given.

For advertising channel packets the format of the device address differs, depending on the selected address type.

Note: This parameter is relevant for advertising event or frame configuration. Refer to [Tx/Rx Address Type - Bluetooth LE Device's Addr Type](#) for information about the available package types for the respective "Controller Roles".

- "Public Address Types"

The public address is given from the registration authority IEEE and is composed of:

 - LSB: 24 bits = company_assigned
 - MSB: 24 bits = company_id
- "Private Address Type"

A private address is optional and composed of:

- LSB: 24 bits = hash
- MSB: 24 bits = random

Remote command:

Company_Assigned and Company_Id in Advertiser's Device Address

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:ACID`

on page 110

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:ACASSigned`

on page 110

Company_Assigned and Company_Id in Scanner's Device Address

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:SCASigned`

on page 110

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:SCID`

on page 110

Company_Assigned and Company_Id in Initiator's Device Address

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:ICASigned`

on page 110

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:ICID`

on page 110

CRC Initial

Sets the initialization value for the CRC (Cyclic Redundary Check, 24 bits) calculation.

A packet has been received correctly, when it has passed the CRC check.

Note: This parameter is relevant for advertising frame configuration and the packet type CONNECT_REQ.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:CIValue`

on page 112

Transmit Window Size

Sets the size of the transmit window, regarding to the start point.

Note that the scan window size is less or equal to the value of the connection interval, see "[Connection Event Interval](#)" on page 32.

Note: This parameter is relevant for advertising frame configuration and for the packet types CONNECT_REQ and CONNECTION_UPDATE_REQ.

This parameter is also indicated in the Frame Configuration Dialog.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:WSize`

on page 122

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:NWSize`

on page 119

Transmit Window Offset

Sets the start point of the transmit window.

Note: This parameter is relevant for advertising frame configuration and for the packet types CONNECT_REQ and CONNECTION_UPDATE_REQ.

This parameter is also indicated in the Frame Configuration Dialog.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:WOffset`
on page 122

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:NWOffset`
on page 118

Sleep Clock Accuracy

Defines the master's clock accuracy with specified encoding. This parameter is used by the slave to determine required listening windows in the LL connection. It is a controller design parameter known by the Controller.

Note: This parameter is relevant for advertising frame configuration and the packet type CONNECT_REQ.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:SCACcuracy`
on page 120

Error Code

Sets the error code value to inform the remote device why the connection is about to be terminated in case of LL_TERMINATE_IND packet. On the other hand, this parameter for LL_REJECT_IND packet is used for the reason a request was rejected. A 8 bit value is set.

Note: This parameter is relevant for data frame configuration and the packet type LL_TERMINATE_IND and LL_REJECT_IND.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:ECODE`
on page 114

Company ID

Sets the company identifier of the manufacturer of the Bluetooth Controller. A 16 bit value is set.

Note: This parameter is relevant for data frame configuration and for the packet type LL_VERSION_IND.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:CID`
on page 111

Version Number

Sets the version of the Bluetooth Controller specification (8 bits).

Note: This parameter is relevant for data frame configuration and the packet type LL_VERSION_IND

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:VNUMBER`
on page 121

Sub Version Number

Sets a unique value for each implementation or revision of an implementation of the Bluetooth Controller.

A 16 bit value is set.

Note: This parameter is relevant for data frame configuration and for the packet type LL_VERSION_IND.

Remote command:

`[ :SOURCE<hw> ] :BB:BT00th:ECONfiguration:PCONfiguration:SVNumber`
on page 121

Graph

The figure in the packet configuration dialog shows the packet structure of the currently selected packet type.

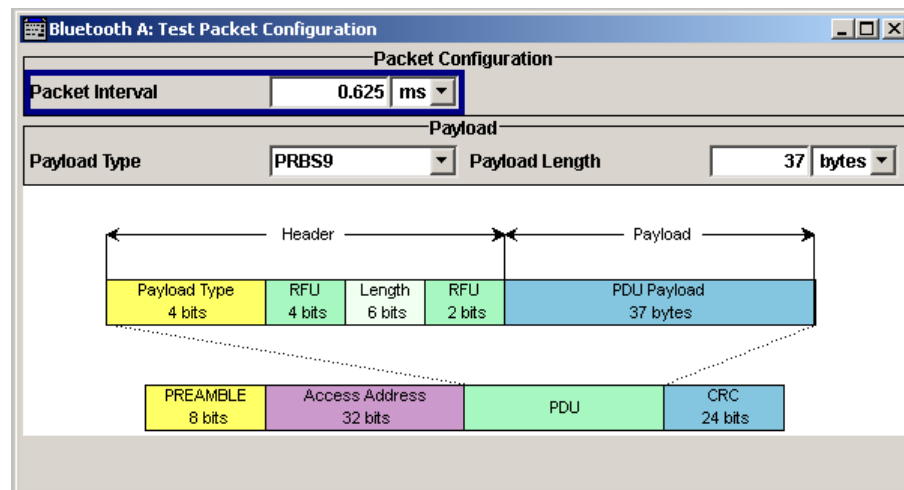
Remote command:

n.a.

3.6.2 Test Packet Configuration Settings

The "Test Packet Configuration" dialog is reached via the "Bluetooth" main dialog, or alternatively the "Dirty Transmitter" test dialog.

The dialog contains the settings, necessary to configure the test packet and graphically shows the distribution of the packets.

**Packet Interval**

Sets the time interval between two consecutive test packets, with regard to the starting points.

Test Packet Interval

Note: This parameter is relevant for test packet types only.

Remote command:

`[ :SOURCE<hw> ] :BB:BT00th:DTTest:TPConfiguration:TPInterval`
on page 106

Payload Type

Selects the data source used for the payload test packets.

Note: This parameter is relevant for test packet types only.

"PRBS 9, 15" Select a PRBS-modulated data sequence (PRBS = pseudo random binary sequence) for testing.

"Pattern 1, 2, 3, 4, 5, 6" Pattern is predefined.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:DTTest:TPConfiguration:UPSource`
on page 106

Payload Length

Sets the payload length.

Note: This parameter is relevant for test packet types only.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:DTTest:TPConfiguration:UPLength`
on page 106

3.7 Dirty Transmitter Test

The Dirty Transmitter settings contain parameters which can be changed for the master signal in order to test the connection under 'dirty transmitter' conditions and define the influence on the receiver quality (bit error rate tests).

Dirty transmitter parameters according to the Bluetooth test specification (Basic Rate) are given in the table below.

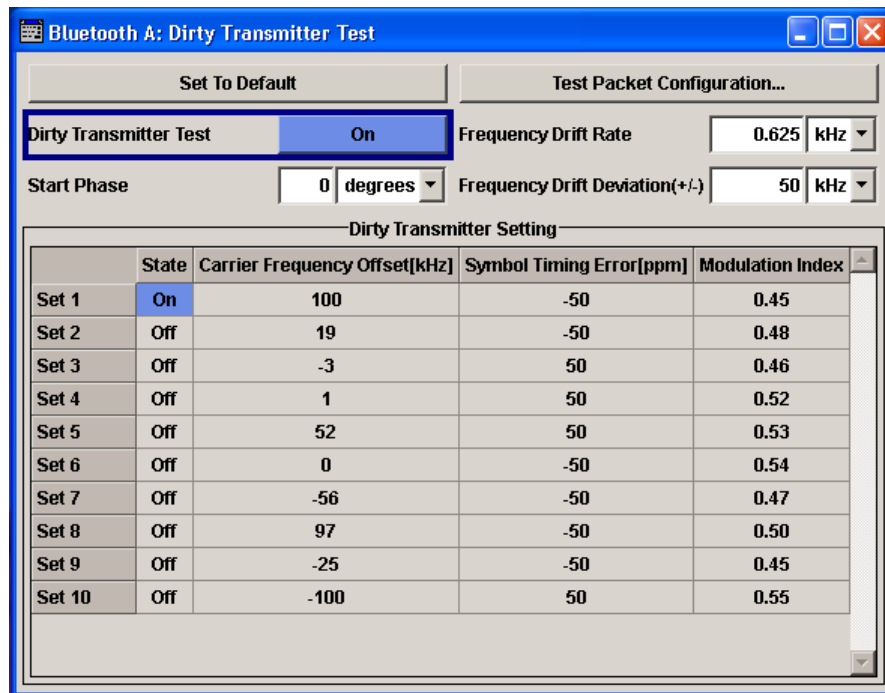
Table 3-3: Dirty transmitter parameters according to the Bluetooth test specification (Basic Rate)

Set	Carrier Frequency Offset	Symbol Timing Error	Modulation Index
1	75	-20	0.28
2	14	-20	0.30
3	-2	+20	0.29
4	1	+20	0.32
5	39	+20	0.33
6	0	-20	0.34
7	-42	-20	0.29
8	74	-20	0.31
9	-19	-20	0.28
10	-75	+20	0.35

Dirty transmitter parameters according to the Bluetooth test specification (EDR) are given in the table below.

Table 3-4: Dirty transmitter parameters according to the Bluetooth test specification (EDR)

Set	Carrier Frequency Offset	Symbol Timing Error
1	0	0
2	+65	-20
3	-65	+20



Provided are the following settings:

Set to Default

Calls the default settings for the Dirty Transmitter Test.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:DTTest:STDefault` on page 100

Test Packet Configuration

Opens the dialog for setting a test packet configuration, see [chapter 3.6.2, "Test Packet Configuration Settings"](#), on page 44.

Dirty Transmitter Test

(Available only for packet types DH1, DH3, DH5, 2-DH1, 2-DH3, 2 - DH5, 3-DH1, 3-DH3, 3-DH5, 2-EV3, 2-EV5, 3-EV3, 3-EV5.)

Activates or deactivates the Dirty Transmitter Test.

For Basic Rate packets, each set of parameters in the "Dirty Transmitter Setting" table below is used for a duration of 20 ms. After 20 ms, the following set is used, continuing with the first set after the sequence is completed.

For EDR packets, the parameter sets apply for 20 packets each.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:DTTest:DTTState` on page 99

Start Phase

Enters a start phase.

The start phase of the sine wave used to drift the modulated Bluetooth signal around center frequency + carrier frequency offset is set here.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:DTTest:SPHase` on page 100

Frequency Drift Rate

Enters a frequency drift rate.

A sine wave is used to drift the modulated Bluetooth signal around center frequency + carrier frequency offset with the set frequency drift rate.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:DTTest:FDRate` on page 100

Frequency Drift Deviation (+/-)

Enters a frequency drift deviation.

A sine wave is used to drift the modulated Bluetooth signal around center frequency + carrier frequency offset. The maximum deviation reached during the drift equals the set frequency drift deviation.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:DTTest:FDDeviation` on page 99

Dirty Transmitter Setting

Indicates the dirty transmitter parameters according to the Bluetooth test specification (Basic Rate).

State ← Dirty Transmitter Setting

Activates or deactivates the corresponding parameter set.

If a set deactivated, its parameters are skipped in the sequence. Instead, the next active set is used.

For Basic Rate packets, each set applies to 20ms of signal. For EDR packets, each set applies to 20 packets.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:DTTest:TABLE:LONG:SET<ch>:STATE`
on page 102

`[ :SOURCE<hw> ] :BB:BT0oth:DTTest:TABLE:SHORT:SET<ch>:STATE`
on page 103

Carrier Frequency Offset kHz ← Dirty Transmitter Setting

Enters a carrier frequency offset.

The center frequency of the modulated RF carrier is offset by the specified value.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:DTTest:TABLE:LONG:SET<ch>:CFOffset`
on page 101

`[ :SOURCE<hw> ] :BB:BT0oth:DTTest:TABLE:SHORT:SET<ch>:CFOffset`
on page 102

Symbol Timing Error ← Dirty Transmitter Setting

Enters the symbol timing error in ppm.

The Symbol Timing Error modifies the symbol clock frequency by the set amount.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:DTTest:TABLE:LONG:SET<ch>:STError`
on page 102

`[ :SOURCE<hw> ] :BB:BT0oth:DTTest:TABLE:SHORT:SET<ch>:STError`
on page 103

Modulation Index ← Dirty Transmitter Setting

(Only for Basic Rate Packets)

Enters the modulation index.

The modulation index specifies the frequency deviation.

The modulation index h is defined as:

$$h = \frac{2\Delta f}{f_{symbol}}$$

where f_{symbol} is the "symbol rate" and Δf is the "frequency deviation".

According to the Bluetooth standard, the modulation index is allowed to vary between 0.28 and 0.35.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:DTTest:TABLE:LONG:SET<ch>:MINdex`
on page 101

3.8 Filter/Clipping Settings

To access this dialog, select "Main Menu > Filter/Clipping".

The baseband filter, sample rate variation and clipping are defined in this menu.

Bluetooth A: Filter/Clipping Settings

Filter

GFSK section

Filter: Gauss(FSK)

B*T: 0.50

Impulse Length: ☒ Auto 10

Oversampling: ☒ Auto 10

n/4 DQPSK section

Filter: Cosine

Roll Off Factor: 0.10

Modulation Settings

Modulation Type: GFSK + n/4 DQPSK

Frequency Deviation: 140.0 kHz

Modulation Index: 0.28

Symbol Rate Variation: 1.000 000 000 MHz

Clipping

State: ☐ On

Clipping Level: 100 %

Clipping Mode: Vector |i+jq|

In the "Filter" section, the settings are made for the baseband filter. The settings for modulation are collected in the "Modulation Settings" section. The settings for clipping are collected in the "Clipping" section.

3.8.1 Filter Settings

Provided are the following settings for configuring the baseband filter:

Filter

Indicates the filter used for GFSK part.

With EDR Packets, you can set the filter used for DQPSK/8DPSK sections.

Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:FILTer:TYPE` on page 71

Roll Off Factor / B xT

Sets the filter parameter.

Sets the filter parameter.

The filter parameter offered ("Roll Off Factor" or "BxT") depends on the currently selected filter type. This parameter is preset to the default for each of the predefined filters.

Remote command:

[:SOURce<hw>] :BB:BT00th:FILTer:PARAmeter:APCO25 on page 73
 [:SOURce<hw>] :BB:BT00th:FILTer:PARAmeter:COsine on page 74
 [:SOURce<hw>] :BB:BT00th:FILTer:PARAmeter:FGAuss on page 74
 [:SOURce<hw>] :BB:BT00th:FILTer:PARAmeter:GAUSs on page 74
 [:SOURce<hw>] :BB:BT00th:FILTer:PARAmeter:PGAuss on page 75
 [:SOURce<hw>] :BB:BT00th:FILTer:PARAmeter:RCOSine on page 75
 [:SOURce<hw>] :BB:BT00th:FILTer:PARAmeter:SPHase on page 75

Cut Off Frequency Factor

(available for filter parameter Lowpass only)

Sets the value for the cut off frequency factor. The cut off frequency of the filter can be adjusted to reach spectrum mask requirements.

Remote command:

[:SOURce<hw>] :BB:BT00th:FILTer:PARAmeter:LPASs on page 74

Impulse Length

(For WinIQSIM2 only)

Displays the number of filter taps. If the check box is activated, the most sensible parameter values are selected. The value depends on the coherence check. If the check box is deactivated, the values can be changed manually.

Remote command:

[:SOURce<hw>] :BB:BT00th:FILTer:ILENgtH:AUTO[:STATe] on page 72
 [:SOURce<hw>] :BB:BT00th:FILTer:ILENgtH on page 71

Oversampling

(For WinIQSIM2 only)

Determines the upsampling factor. If the check box is activated, the most sensible parameter values are selected. The value depends on the coherence check. If the check box is deactivated, the values can be changed manually.

Remote command:

[:SOURce<hw>] :BB:BT00th:FILTer:OSAMpling:AUTO[:STATe] on page 72
 [:SOURce<hw>] :BB:BT00th:FILTer:OSAMpling on page 72

3.8.2 Modulation Settings

Provided are the following settings:

Modulation type

Displays the modulation type used for the current packet selection.

Remote command:

[:SOURce<hw>] :BB:BT00th:FILTer:MTYPE on page 73

Frequency deviation

Enter the frequency deviation of the frequency modulated part.

The frequency deviation can be varied in a range from 100.0 kHz to 200.0 kHz according to Bluetooth specification.

Remote command:

[\[:SOURce<hw>\]:BB:BT00th:MSEttings:FDEVIation](#) on page 73

Modulation index

Displays the modulation index resulting from the entered frequency deviation value.

Modulation index is calculated from the given frequency deviation and symbol rate values.

The modulation index h is defined as:

$$h = \frac{2\Delta f}{f_{symbol}}$$

where f_{symbol} is the "symbol rate" and Δf is the "frequency deviation".

According to the Bluetooth standard, the modulation index is allowed to vary between 0.28 and 0.35.

Remote command:

[\[:SOURce<hw>\]:BB:BT00th:FILTer:MINdex](#) on page 73

Symbol Rate Variation

Enter the symbol rate.

Remote command:

[\[:SOURce<hw>\]:BB:BT00th:SRATe:VARiation](#) on page 76

3.8.3 Clipping Settings

The settings for clipping are collected in the "Clipping" section.

Provided are the following settings:

Clipping State

Switches baseband clipping on and off.

Switches baseband clipping on and off.

Baseband clipping is a very simple and effective way of reducing the crest factor of the signal. Since clipping is done prior to filtering, the procedure does not influence the spectrum. The EVM however increases.

Remote command:

[\[:SOURce<hw>\]:BB:BT00th:CLIPping:STATe](#) on page 71

Clipping Level

Sets the limit for clipping.

This value indicates at what point the signal is clipped. It is specified as a percentage, relative to the highest level. 100% indicates that clipping does not take place.

Remote command:

`[ :SOURCE<hw> ] :BB:BT00th:CLIPping:LEVel` on page 70

Clipping Mode

Selects the clipping method. A graphic illustration of the way in which these two methods work is given in the menu.

Selects the clipping method. A graphic illustration of the way in which these two methods work is given in the dialog.

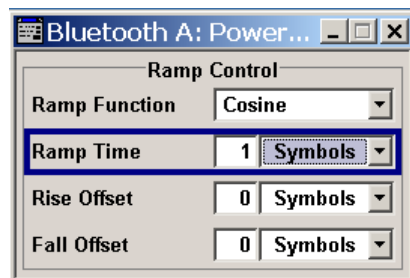
- "Vector $|i + q|$ "
The limit is related to the amplitude $|i + q|$. The I and Q components are mapped together, the angle is retained.
- "Scalar $|i| + |q|$ "
The limit is related to the absolute maximum of all the I and Q values $|i| + |q|$. The I and Q components are mapped separately, the angle changes.

Remote command:

`[ :SOURCE<hw> ] :BB:BT00th:CLIPping:MODE` on page 70

3.9 Power Ramping Settings

To access this dialog, select "Main Menu > Power Ramping".



The dialog is used to set the power ramping.

Provided are the following settings:

Ramp Function

Selects the form of the transmitted power, i.e. the shape of the rising and falling edges during power ramp control.

- "Linear" The transmitted power rises and falls with linear fashion.
- "Cosine" The transmitted power rises and falls with a cosine-shaped edge. This gives rise to a more favorable spectrum than the Linear setting.

Remote command:

`[ :SOURCE<hw> ] :BB:BT00th:PRAMping:RFUNction` on page 89

Ramp Time

Sets the power ramping rise time and fall time for a burst.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:PRAMping:RTIME` on page 90

Rise Offset

Sets the offset in the rising edge of the envelope at the start of a burst. A positive value moves the ramp into the beginning of a transmitted packet and a negative value introduces an additional guard period before the start of the packet.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:PRAMping:ROFFset` on page 89

Fall Offset

Sets the offset of the falling edge of the envelope at the end of a burst. A positive value introduces a guard period after the end of the packet and a negative value moves the ramp into the end part of the transmitted packet.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:PRAMping:FOFFset` on page 89

3.10 Trigger/Marker/Clock Settings



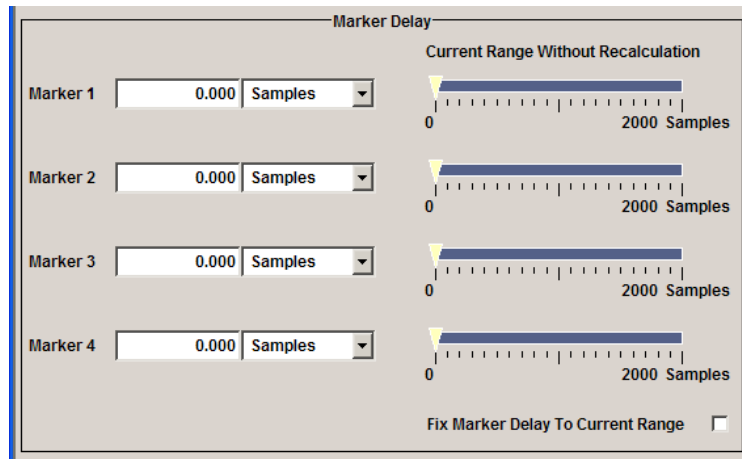
The trigger, clock, and marker delay functions are available for R&S SMx and R&S AMU instruments only.

To access this dialog, select "Main Menu > Trigger/Marker".

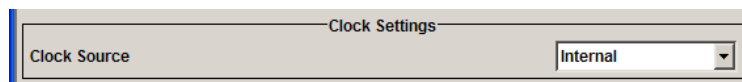
The "Trigger In" section is where the trigger for the signal is set. Various parameters will be provided for the settings, depending on which trigger source - internal or external - is selected. The current status of signal generation ("Running" or "Stopped") is indicated for all trigger modes.

The "Marker Mode" section is where the marker signals at the MARKER output connectors are configured.

The "Marker Delay" section is where a marker signal delay can be defined, either without restriction or restricted to the dynamic section, i.e., the section in which it is possible to make settings without restarting signal and marker generation.



The "Clock Settings" section is where the clock source is selected and - in the case of an external source - the clock type.



The buttons in the last section lead to submenu for general trigger, clock and mapping settings.



3.10.1 Trigger Settings

The "Trigger In" section is where the trigger for the signal is set. Various parameters will be provided for the settings, depending on which trigger source - internal or external - is selected. The current status of signal generation ("Running" or "Stopped") is indicated for all trigger modes.

Trigger Mode

Selects trigger mode, i.e. determines the effect of a trigger event on the signal generation.

- "Auto"
The signal is generated continuously.
- "Retrigger"
The signal is generated continuously. A trigger event (internal or external) causes a restart.
- "Armed_Auto"
The signal is generated only when a trigger event occurs. Then the signal is generated continuously.

An "Arm" stops the signal generation. A subsequent trigger event (internal with or external) causes a restart.

- "Armed_Retrigger"
The signal is generated only when a trigger event occurs. Then the signal is generated continuously. Every subsequent trigger event causes a restart.
An "Arm" stops signal generation. A subsequent trigger event (internal with or external) causes a restart.
- "Single"
The signal is generated only when a trigger event occurs. Then the signal is generated once to the length specified at "Signal Duration".
Every subsequent trigger event (internal or external) causes a restart.

Remote command:

[\[:SOURce<hw>\]:BB:BT00th\[:TRIGger\]:SEquence](#) on page 81

Signal Duration Unit

Selects the unit for the entry of the length of the signal sequence to be output in the Single trigger mode. Available units are sequence length (SL) or frames.

"Sequence Length" The selected unit for the entry of the length of the signal sequence at the output in the Single trigger mode is sequence length .

"Frames" The selected unit for the entry of the length of the signal sequence at the output in the Single trigger mode is frame.

Remote command:

[\[:SOURce<hw>\]:BB:BT00th:TRIGger:SLUNit](#) on page 79

Signal Duration

Enters the length of the signal sequence to be output in the "Single" trigger mode.

Use this parameter to deliberately output part of the signal, an exact sequence of the signal, or a defined number of repetitions of the signal.

Remote command:

[\[:SOURce<hw>\]:BB:BT00th:TRIGger:SELength](#) on page 79

Running/Stopped

For enabled modulation, displays the status of signal generation for all trigger modes.

- "Running"
The signal is generated; a trigger was (internally or externally) initiated in triggered mode.
- "Stopped"
The signal is not generated and the instrument waits for a trigger event.

Remote command:

[\[:SOURce<hw>\]:BB:BT00th:TRIGger:RMODe](#) on page 78

Arm

Stops signal generation manually. This button appears only with "Running" signal generation in the "Armed_Auto" and "Armed_Retrigger" trigger modes.

Signal generation can be restarted by a new trigger (internally with "Execute Trigger" or externally).

Remote command:

`[ :SOURce<hw> ] :BB:BT00th:TRIGger:ARM:EXECute` on page 76

Execute Trigger

Executes a trigger manually. A manual trigger can be executed only if an internal trigger source and a trigger mode other than "Auto" have been selected.

Remote command:

`[ :SOURce<hw> ] :BB:BT00th:TRIGger:EXECute` on page 77

Trigger Source

Selects trigger source. This setting is effective when a trigger mode other than "Auto" has been selected.

- "Internal"
The trigger event is executed by "Execute Trigger".
- "Internal (Baseband A/B)"
(two-path instruments)
The trigger event is the trigger signal from the second path
- "External (Trigger 1/2)"
The trigger event is the active edge of an external trigger signal, supplied at the TRIGGER 1/2 connector.
Use the "Global Trigger/Clock Settings" dialog to define the polarity, the trigger threshold and the input impedance of the trigger signal.

Remote command:

`[ :SOURce<hw> ] :BB:BT00th:TRIGger:SOURce` on page 80

Sync. Output to External Trigger

(enabled for Trigger Source External)

Enables/disables output of the signal synchronous to the external trigger event.

For R&S SMBV instruments:

For one or two or more R&S SMBVs configured to work in a master-slave mode for synchronous signal generation, configure this parameter depending on the provided system trigger event and the properties of the output signal. See the table below for an overview of the required settings.

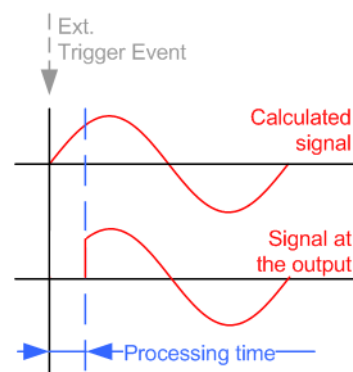
Table 3-5: Typical Applications

System Trigger	Application	"Sync. Output to External Trigger"
Common External Trigger event for the master and the slave instruments	All instruments are synchronous to the external trigger event	ON
	All instruments are synchronous among themselves but starting the signal from first symbol is more important than synchronicity with external trigger event	OFF
Internal trigger signal of the master R&S SMBV for the slave instruments	All instruments are synchronous among themselves	OFF

"On"

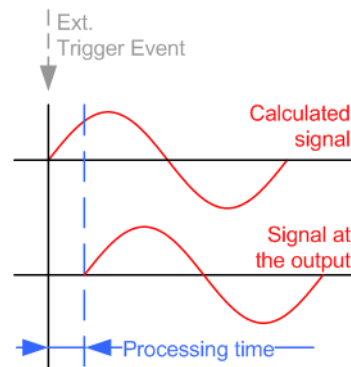
Corresponds to the default state of this parameter.

The signal calculation starts simultaneously with the external trigger event but because of the instrument's processing time the first samples are cut off and no signal is outputted. After elapsing of the internal processing time, the output signal is synchronous to the trigger event.



"Off"

The signal output begins after elapsing of the processing time and starts with sample 0, i.e. the complete signal is outputted. This mode is recommended for triggering of short signal sequences with signal duration comparable with the processing time of the instrument.



Remote command:

`[ :SOURce<hw> ] :BB:BT00th:TRIGger[:EXtErnal]:SYNChronize:OUTPut`
on page 77

Trigger Delay

Delays the trigger event of the signal from:

- the external trigger source
- the other path

Use this setting to:

- synchronize the instrument with the device under test (DUT) or other external devices

Remote command:

`[ :SOURce<hw> ] :BB:BT00th:TRIGger[:EXtErnal<ch>]:DELay` on page 80
`[ :SOURce<hw> ] :BB:BT00th:TRIGger:OBASeband:DELay` on page 78

Trigger Inhibit

Sets the duration for inhibiting a new trigger event subsequent to triggering. The input is to be expressed in samples.

In the "Retrigger" mode, every trigger signal causes signal generation to restart. This restart is inhibited for the specified number of samples.

This parameter is only available on external triggering or on internal triggering via the second path.

For two-path instruments, the trigger inhibit can be set separately for each of the two paths.

Remote command:

`[ :SOURce<hw> ] :BB:BT00th:TRIGger[:EXtErnal<ch>]:INHibit` on page 81
`[ :SOURce<hw> ] :BB:BT00th:TRIGger:OBASeband:INHibit` on page 78

3.10.2 Marker Mode

The marker output signal for synchronizing external instruments is configured in the marker settings section "Marker Mode".

The R&S SMBV supports only two markers.

Marker Mode

Selects a marker signal for the associated MARKER output.

- | | |
|---------------------|---|
| "Restart" | A marker signal is generated at the start of each signal sequence. |
| "Frame Start" | A marker signal is generated at the start of each frame. |
| "Frame Active Part" | The marker masks the active part of the frame. At the start of each burst, the marker signal changes to high. It changes back to low after the end of each burst. |
| "Pulse" | A regular marker signal is generated. The pulse frequency is defined by entering a divider. The frequency is derived by dividing the sample rate by the divider. The input box for the divider opens when "Pulse" is selected, and the resulting pulse frequency is displayed below it. |

Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:TRIGger:OUTPut<ch>:PULSe:DIVider`

on page 85

`[ :SOURce<hw> ] :BB:BT0oth:TRIGger:OUTPut<ch>:PULSe:FREQuency?`

on page 86

- | | |
|------------|---|
| "Pattern " | A marker signal that is defined by a bit pattern is generated. The pattern has a maximum length of 32 bits and is defined in an input field which opens when pattern is selected. |
|------------|---|

Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:TRIGger:OUTPut<ch>:PATtern` on page 85

- | | |
|-----------------|---|
| "ON/OFF Period" | A regular marker signal that is defined by an ON/OFF ratio is generated. A period lasts one ON and OFF cycle. The "ON Time" and "OFF Time" are each expressed as a number of samples and are set in an input field which opens when ON/OFF ratio is selected. |
|-----------------|---|



Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:TRIGger:OUTPut<ch>:ONTime` on page 85

`[ :SOURce<hw> ] :BB:BT0oth:TRIGger:OUTPut<ch>:OFFTime` on page 85

Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:TRIGger:OUTPut<ch>:MODE` on page 84

3.10.3 Marker Delay

The delay of the signals on the MARKER outputs is set in the "Marker Delay" section.



The marker delay functions are available for R&S SMx and R&S AMU instruments only.

The R&S SMBV supports only two markers.

Marker x Delay

Enters the delay between the marker signal at the marker outputs and the start of the frame or slot.

The input is expressed as a number of symbols/samples. If the setting "Fix marker delay to dynamic range" is enabled, the setting range is restricted to the dynamic range. In this range the delay of the marker signals can be set without restarting the marker and signal.

Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:TRIGger:OUTPut<ch>:DELay` on page 83

Current Range without Recalculation

Displays the dynamic range within which the delay of the marker signals can be set without restarting the marker and signal.

The delay can be defined by moving the setting mark.

Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:TRIGger:OUTPut<ch>:DELay:MINimum?`

on page 83

`[ :SOURce<hw> ] :BB:BT0oth:TRIGger:OUTPut<ch>:DELay:MAXimum?`

on page 83

Fix marker delay to current range

Restricts the marker delay setting range to the dynamic range. In this range the delay can be set without restarting the marker and signal.

Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:TRIGger:OUTPut:DELay:FIXed` on page 82

3.10.4 Clock Settings

The Clock Settings is used to set the clock source and a delay if required.



The clock functions are available for R&S SMx and R&S AMU instruments only.

Sync. Mode

(for R&S SMBV only)

Selects the synchronization mode.

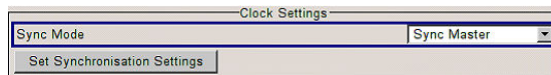
This parameter is used to enable generation of very precise synchronous signal of several connected R&S SMBVs.

Note: If several instruments are connected, the connecting cables from the master instrument to the slave one and between each two consecutive slave instruments must have the same length and type.

Avoid unnecessary cable length and branching points.

"None" The instrument is working in stand-alone mode.

"Sync. Master" The instrument provides all connected instrument with its synchronisation (including the trigger signal) and reference clock signal.



"Sync. Slave" The instrument receives the synchronisation and reference clock signal from another instrument working in a master mode.

Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:CLOCK:SYNChronization:MODE` on page 88

Set Synchronization Settings

(for R&S SMBV only)

Performs automatic adjustment of the instrument's settings required for the synchronization mode, selected with the parameter "Synchronization Mode".

Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:CLOCK:SYNChronization:EXECute` on page 88

Clock Source

Selects the clock source.

"Internal" The internal clock reference is used to generate the symbol clock.

"External" The external clock reference is fed in as the symbol clock or multiple thereof via the CLOCK connector.
The symbol rate must be correctly set to an accuracy of $\pm 2\%$ (see data sheet).
The polarity of the clock input can be changed with the aid of "Global Trigger/Clock Settings".
In the case of two-path instruments this selection applies to path A.

Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:CLOCK:SOURce` on page 87

Clock Mode

(This feature is available for the external clock source only.)

Selects the type of externally supplied clock.

"Sample" A sample clock is supplied via the CLOCK connector.

"Multiple Sample" A multiple of the sample clock is supplied via the CLOCK connector.
The sample clock is derived internally from this. The value range is 1 to 64.
The "Clock Multiplier" field provided allows the multiplication factor to be entered.

Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:CLOCK:MODE` on page 86

Clock Multiplier

(This feature is available for the external clock source only.)

Enters the multiplication factor for clock type Multiple Sample.

Remote command:

`[ :SOURce<hw> ] :BB:BT00th:CLOCK:MULTiplier` on page 87

Measured External Clock

Provided for permanent monitoring of the enabled and externally supplied clock signal.

Remote command:

`CLOCK:INPut:FREQuency?`

3.10.5 Global Settings

The buttons in this section lead to dialogs for general trigger, clock and mapping settings.

These settings are available for R&S SMx and R&S AMU instruments only.

Global Trigger/Clock Settings

Calls the "Global Trigger/Clock/Input Settings" dialog.

This dialog is used among other things for setting the trigger threshold, the input impedance and the polarity of the clock and trigger inputs.

In the case of two-path instruments, these settings are valid for both paths.

The parameters in this dialog affect all digital modulations and standards, and are described in chapter "Global Trigger/Clock/Input Settings" in the Operating Manual.

User Marker / AUX I/O Settings

Calls the "User Marker AUX I/O Settings" dialog, used to map the connector on the rear of the instruments.

See also "User Marker / AUX I/O Settings" in the Operating Manual.

4 Remote-Control Commands

The following commands are required to perform signal generation with the Bluetooth options in a remote environment. We assume that the R&S Signal Generator has already been set up for remote operation in a network as described in the R&S Signal Generator documentation. A knowledge about the remote control operation and the SCPI command syntax are assumed.



Conventions used in SCPI command descriptions

For a description of the conventions used in the remote command descriptions, see section "Remote Control Commands" in the R&S Signal Generator operating manual.

The commands in the `SOURCE:BB:BT00th` subsystem are described in three sections, separated into general remote commands, commands for Packet Configuration settings and commands for Dirty Transmitter Test settings.

This subsystem contains commands for the primary and general settings of the Bluetooth standard. These settings concern activation and deactivation of the standard, setting filter, clock, trigger and clipping settings, defining the symbol rate variation and the sequence length, as well as the preset and power adjust setting.

Common Suffixes

The following common suffixes are used in remote commands:

Suffix	Value range	Description
<code>SOURCE<hw></code>	<code>[1] 2</code>	available baseband signals
<code>OUTPut<ch></code>	<code>1 .. 4</code>	available markers R&S SMBV supports two markers
<code>EXTErnal<ch></code>	<code>1 2</code>	external trigger connectors

Placeholder <root>

For commands that read out or save files in the default directory, the default directory is set using command `MMEM:CDIRECTory`. The examples in this description use the place holder `<root>` in the syntax of the command.

- `D:\` - for selecting the internal hard disk of a Windows instrument
- `E:\` - for selecting the memory stick which is inserted at the USB interface of a Windows instrument
- `/var/user/` - for selecting the internal flash card of a Linux instrument
- `/usb/` - for selecting the memory stick which is inserted at the USB interface of a Linux instrument.



Tasks (in manual or remote operation) that are also performed in the base unit in the same way are not described here.

In particular, this includes:

- Managing settings and data lists, i.e. storing and loading settings, creating and accessing data lists, accessing files in a particular directory, etc.
- Information on regular trigger, marker and clock signals as well as filter settings, if appropriate.
- General instrument configuration, such as configuring networks and remote operation
- Using the common status registers

For a description of such tasks, see the R&S Signal Generator operating manual.

The following commands specific to the Bluetooth are described here:

4.1 General Commands

<code>[SOURce<hw>]:BB:BT00th:BCText?</code>	64
<code>[SOURce<hw>]:BB:BT00th:CTYPE</code>	65
<code>[SOURce<hw>]:BB:BT00th:PRESet</code>	65
<code>[SOURce<hw>]:BB:BT00th:PTYPE</code>	66
<code>[SOURce<hw>]:BB:BT00th:SETTing:CATalog</code>	66
<code>[SOURce<hw>]:BB:BT00th:SETTing:DELeTe</code>	66
<code>[SOURce<hw>]:BB:BT00th:SETTing:LOAD</code>	67
<code>[SOURce<hw>]:BB:BT00th:SETTing:STORe</code>	67
<code>[SOURce<hw>]:BB:BT00th:SETTing:STORe:FAST</code>	67
<code>[SOURce<hw>]:BB:BT00th:SLENgth</code>	68
<code>[SOURce<hw>]:BB:BT00th:STATe</code>	68
<code>[SOURce<hw>]:BB:BT00th:TMODe</code>	68
<code>[SOURce<hw>]:BB:BT00th:VERSIon</code>	69
<code>[SOURce<hw>]:BB:BT00th:WAVeform:CREate</code>	69

`[SOURce<hw>]:BB:BT00th:BCText?`

Queries the state/roll of the controller.

Return values:

<BcText>

string

Connected

(for data channel type)

Shows that the state is Connected.

Advertiser

(for advertising channel type)

the current bluetooth Controller Role is Advertiser
for all ADV-xxx packet types and SCAN_RSP.**Scanner**

(for advertising channel type)

the current bluetooth Controller Role is Scanner
for SCAN_REQ packet type**Initiator**

(for advertising channel type)

the current bluetooth Controller Role is Initiator
for CONNECT_REQ packet type**Example:**

SOUR:BB:BTO:BCT?

Queries the state/roll of the controller.

Usage:

Query only

Manual operation: See ["Bluetooth Controller State"](#) on page 28**[:SOURce<hw>]:BB:BTOoth:CTYPE <CType>**

Determines the channel type. Advertising and data are available.

Parameters:

<CType>

ADVertising | DATA

ADVertising

Selects channel type Advertising.

DATA

Selects channel type Data. Devices in a connected state transmit data channel packets in connection events with a start point and an interval.

*RST: ADVertising

Example:

SOUR:BB:BTO:CTYP ADV

channel type Advertising.

SOUR:BB:BTO:CTYP DATA

channel type Data.

Manual operation: See ["Channel Type"](#) on page 26**[:SOURce<hw>]:BB:BTOoth:PRESet**

The command produces a standardized default for the Bluetooth standard. The settings correspond to the *RST values specified for the commands.

All Bluetooth settings are preset.

Example: BB:BTO:PRES
resets all the Bluetooth settings to default values.

Usage: Event

Manual operation: See ["Set To Default"](#) on page 16

[:SOURce<hw>]:BB:BTOoth:PTYPe <PType>

The available packets depend on the selected transport mode. All packet types as defined in the Bluetooth specifications are supported.

Parameters:

<PType> ID | NULL | POLL | FHS | DM1 | DH1 | DM3 | DH3 | DM5 | DH5 |
AUX1 | ADH1 | ADH3 | ADH5 | AEDH1 | AEDH3 | AEDH5 |
HV1 | HV2 | HV3 | DV | EV3 | EV4 | EV5 | EEV3 | EEV5 |
EEEV3 | EEEV5

*RST: DH1

Example: BB:BTO:PTYP NULL
sets the packet type.

Manual operation: See ["Packet Type"](#) on page 20

[:SOURce<hw>]:BB:BTOoth:SETTing:CATalog <Catalog>

This command reads out the files with Bluetooth settings in the default directory. The default directory is set using command `MMEM:CDIRectory`. Only files with the file extension `*.bto` will be listed.

Parameters:

<Catalog> string

Example: MMEM:CDIR '<root>bluetooth'
sets the default directory to <root>bluetooth.
BB:BTO:SETT:CAT?
reads out all the files with Bluetooth settings in the default directory.
Response: 'Bluetooth_EDR', 'Bluetooth_SCO'
the files Bluetooth_EDR and Bluetooth_SCO are available.

Manual operation: See ["Save/Recall..."](#) on page 16

[:SOURce<hw>]:BB:BTOoth:SETTing:DELeTe <Filename>

This command deletes the selected file with Bluetooth settings. The directory is set using command `MMEM:CDIRectory`. A path can also be specified, in which case the files in the specified directory are read. The file extension may be omitted. Only files with the file extension `*.bto` will be deleted.

Parameters:

<Filename> string

Example:

BB:BTO:SETT:DEL '<root>bluetooth'
deletes the specified file with Bluetooth settings.

Manual operation: See ["Save/Recall..."](#) on page 16

[[:SOURce<hw>]:BB:BTOoth:SETTing:LOAD <Filename>

This command loads the selected file with Bluetooth settings. The directory is set using command `MMEM:CDIRectory`. A path can also be specified, in which case the files in the specified directory are read. The file extension may be omitted. Only files with the file extension `*.bto` will be loaded.

Parameters:

<Filename> string

Example:

BB:BTO:SETT:LOAD 'bluetooth_1'
loads file `bluetooth_1`.

Manual operation: See ["Save/Recall..."](#) on page 16

[[:SOURce<hw>]:BB:BTOoth:SETTing:STORe <Filename>

This command stores the current Bluetooth settings into the selected file. The directory is set using command `MMEM:CDIRectory`. A path can also be specified, in which case the files in the specified directory are read. Only the file name has to be entered. Bluetooth settings are stored as files with the specific file extensions `*.bto`.

Setting parameters:

<Filename> string

Example:

BB:BTO:SETT:STOR 'bluetooth_1'
stores the current Bluetooth settings into file `bluetooth_1`.

Usage:

Setting only

Manual operation: See ["Save/Recall..."](#) on page 16

[[:SOURce<hw>]:BB:BTOoth:SETTing:STORe:FAST <Fast>

Determines whether the instrument performs an absolute or a differential storing of the settings.

Enable this function to accelerate the saving process by saving only the settings with values different to the default ones.

Note: This function is not affected by the "Preset" function.

Parameters:

<Fast> 0 | 1 | OFF | ON

*RST: 1

Manual operation: See ["Save/Recall..."](#) on page 16

[:SOURce<hw>]:BB:BTOoth:SLENgth <SLength>

The command sets the sequence length of the Bluetooth signal in number of frames. This signal is calculated in advance and output in the arbitrary waveform generator.

Parameters:

<SLength> float
Range: 1 to 53687
*RST: 1

Example: BB:BTO:SLEN 10
sets the sequence length to 10 frames.

Manual operation: See ["Sequence Length"](#) on page 20

[:SOURce<hw>]:BB:BTOoth:STATe <State>

The command activates modulation in accordance with the Bluetooth standard. Activating this standard deactivates all the other digital standards and digital modulation modes (in case of two-path instruments, this affects the same path).

BB:BTO:STAT ON deactivates the other standards and digital modulation.

Parameters:

<State> 0 | 1 | OFF | ON
*RST: OFF

Example: BB:BTO:STAT ON
activates modulation in accordance with the Bluetooth standard.

Manual operation: See ["State"](#) on page 16

[:SOURce<hw>]:BB:BTOoth:TMODe <TMode>

The command selects the transport mode.

Parameters:

<TMode> ACL | SCO | ESCO

ACL

The selected transport mode is used for a point-to-point multi-point link establishment between the master and all the slaves participating on the piconet.

SCO

The selected transport mode is used for a point-to-point link establishment between a master and a single slave in the piconet.

ESCO

The selected transport mode is used for a symmetric or asymmetric point-to-point link establishment between a master and a specific slave.

*RST: ACL

Example:

```
BB:BTO:TMOD ACL
selects transport mode ACL.
```

Manual operation: See ["Transport Mode"](#) on page 20**[[:SOURce<hw>]:BB:BTOoth:VERSion <Version>**

The command queries the version of the Bluetooth standard underlying the definitions.

Parameters:

<Version> string

Example:

```
BB:BTO:VERS?
queries the Bluetooth version.
Response: Version 2.1+EDR
```

Manual operation: See ["Bluetooth Version"](#) on page 18**[[:SOURce<hw>]:BB:BTOoth:WAVEform:CREate <Filename>**

This command creates a waveform using the current settings of the "Bluetooth" menu. The file name is entered with the command. The file is stored with the predefined file extension *.wv. The file name and the directory it is stored in are user-definable.

Setting parameters:

<Filename> string

Example:

```
MMEM:CDIR '<root>waveform'
sets the default directory to <root>waveform.
BB:BTO:WAV:CRE 'bluetooth_1'
creates the waveform file bluetooth.wv in the default directory.
```

Usage: Setting only**Manual operation:** See ["Generate Waveform File..."](#) on page 18

4.2 Filter/Clipping Settings

<code>[SOURce<hw>]:BB:BTOoth:CLIPping:LEVel</code>	70
<code>[SOURce<hw>]:BB:BTOoth:CLIPping:MODE</code>	70
<code>[SOURce<hw>]:BB:BTOoth:CLIPping:STATe</code>	71
<code>[SOURce<hw>]:BB:BTOoth:FILTer:TYPE</code>	71
<code>[SOURce<hw>]:BB:BTOoth:FILTer:ILENgtH</code>	71
<code>[SOURce<hw>]:BB:BTOoth:FILTer:ILENgtH:AUTO[:STATe]</code>	72
<code>[SOURce<hw>]:BB:BTOoth:FILTer:OSAMpling</code>	72
<code>[SOURce<hw>]:BB:BTOoth:FILTer:OSAMpling:AUTO[:STATe]</code>	72
<code>[SOURce<hw>]:BB:BTOoth:FILTer:MINDex</code>	73
<code>[SOURce<hw>]:BB:BTOoth:FILTer:MTYPE</code>	73
<code>[SOURce<hw>]:BB:BTOoth:MSETtings:FDEVIation</code>	73
<code>[SOURce<hw>]:BB:BTOoth:FILTer:PARAmeter:APCO25</code>	73
<code>[SOURce<hw>]:BB:BTOoth:FILTer:PARAmeter:COSSine</code>	74
<code>[SOURce<hw>]:BB:BTOoth:FILTer:PARAmeter:FGAuss</code>	74
<code>[SOURce<hw>]:BB:BTOoth:FILTer:PARAmeter:GAUSS</code>	74
<code>[SOURce<hw>]:BB:BTOoth:FILTer:PARAmeter:LPASS</code>	74
<code>[SOURce<hw>]:BB:BTOoth:FILTer:PARAmeter:PGAuss</code>	75
<code>[SOURce<hw>]:BB:BTOoth:FILTer:PARAmeter:RCOSSine</code>	75
<code>[SOURce<hw>]:BB:BTOoth:FILTer:PARAmeter:SPHase</code>	75
<code>[SOURce<hw>]:BB:BTOoth:SRATe:VARiation</code>	76

`[SOURce<hw>]:BB:BTOoth:CLIPping:LEVel <Level>`

The command sets the limit for level clipping (Clipping). This value indicates at what point the signal is clipped. It is specified as a percentage, relative to the highest level. 100% indicates that clipping does not take place.

Parameters:

<code><Level></code>	float
Range:	1 PCT to 100 PCT
Increment:	1 PCT
*RST:	100 PCT

Example:

```
BB:BTO:CLIP:LEV 80
sets the limit for level clipping to 80% of the maximum level.
BB:BTO:CLIP:STAT ON
activates level clipping.
```

Manual operation: See "[Clipping Level](#)" on page 51

`[SOURce<hw>]:BB:BTOoth:CLIPping:MODE <Mode>`

The command sets the method for level clipping (Clipping).

Parameters:

<Mode> VECTOR | SCALAR

VECTORThe reference level is the amplitude $|i+jq|$.**SCALAR**

The reference level is the absolute maximum of the I and Q values.

*RST: VECTOR

Example:

```
BB:BTO:CLIP:MODE VECT
```

 sets the amplitude as reference level.
Manual operation: See ["Clipping Mode"](#) on page 52**[[:SOURce<hw>]:BB:BTOoth:CLIPping:STATe <State>**

The command activates level clipping (Clipping). The value is defined with the command `BB:BTO:CLIPping:LEVel`, the mode of calculation with the command `BB:BTO:CLIPping:MODE`.

Parameters:

<State> 0 | 1 | OFF | ON

*RST: OFF

Example:

```
BB:BTO:CLIP:STAT ON
```

 activates level clipping.
Manual operation: See ["Clipping State"](#) on page 51**[[:SOURce<hw>]:BB:BTOoth:FILTer:TYPE <Type>**

The command selects the filters used for $\pi/4$ DQPSK and 8DPSK modulations. This opens a selection window containing all the filters available to the instrument.

Parameters:<Type> RCOSine | COSine | GAUSs | LGAuss | CONE | COF705 |
COEQUALizer | COFEQUALizer | C2K3x | APCO25 | SPHase |
RECTangle | PGAuss | LPASs | DIRac | ENPSHape |
EWPSHape

*RST: RCOS

Example:

```
BB:BTO:FILT:TYPE RCOS
```

 sets the filter type RCOSine.
Manual operation: See ["Filter"](#) on page 49**[[:SOURce<hw>]:BB:BTOoth:FILTer:ILENgtH <ILength>**

The command sets the impulse length (the number of filter taps).

Parameters:

<ILength> float
 Range: 1 to 128
 Increment: 1
 *RST: 10

Example:

BB:BTO:FILT:ILEN 10
 sets the number of filter tabs to 10.

Manual operation: See ["Impulse Length"](#) on page 50

[<SOURce<hw>]:BB:BTOoth:FILT:ILENgtH:AUTO[<STATe>] <State>

The command activates/deactivates the impulse length state. If activated, the most sensible parameter values are selected. The value depends on the coherence check.

Parameters:

<State> 0 | 1 | OFF | ON
 *RST: ON

Example:

BB:BTO:FILT:ILEN:AUTO ON
 the most sensible parameters are selected automatically.

Manual operation: See ["Impulse Length"](#) on page 50

[<SOURce<hw>]:BB:BTOoth:FILT:OSAMplng <OSampling>

The command sets the upsampling factor.

Parameters:

<OSampling> integer
 Range: 1 to 32
 *RST: 10

Example:

BB:BTO:FILT:OSAM 10
 sets the upsampling factor to 10.

Manual operation: See ["Oversampling"](#) on page 50

[<SOURce<hw>]:BB:BTOoth:FILT:OSAMplng:AUTO[<STATe>] <State>

The command activates/deactivates the upsampling factor state. If activated, the most sensible parameter values are selected. The value depends on the coherence check. If deactivated, the values can be changed manually.

Parameters:

<State> 0 | 1 | OFF | ON
 *RST: ON

Example:

BB:BTO:FILT:OSAM:AUTO ON
 the most sensible parameters are selected automatically.

Manual operation: See ["Oversampling"](#) on page 50

[[:SOURce<hw>]:BB:BTOoth:FILTer:MINdex <MIndex>

Queries the modulation index resulting from the entered frequency deviation value.

Parameters:

<MIndex> string

Example:

BB:BTO:FILT:MIND

Queries the modulation index

Response: 0.5

Manual operation: See "[Modulation index](#)" on page 51

[[:SOURce<hw>]:BB:BTOoth:FILTer:MTYPE <MType>

Queries the modulation type used for the current packet selection.

Parameters:

<MType> string

Example:

BB:BTO:FILT:MTYP?

Queries the modulation type

Manual operation: See "[Modulation type](#)" on page 50

[[:SOURce<hw>]:BB:BTOoth:MSETtings:FDEViation <FDeviation>

The frequency deviation can be varied in a range from 100.0 kHz to 200.0 kHz.

Parameters:

<FDeviation> float

Range: 100 kHz to 200 kHz

Increment: 0.1 kHz

*RST: 160 kHz

Example:

BB:BTO:MSET:FDEV 160

sets a frequency deviation.

Manual operation: See "[Frequency deviation](#)" on page 51

[[:SOURce<hw>]:BB:BTOoth:FILTer:PARAmeter:APCO25 <Apco25>

The command sets the roll-off factor for filter type APCO25.

Parameters:

<Apco25> float

Range: 0.05 to 0.99

Increment: 0.01

*RST: 0.2

Example:

BB:BTO:FILT:PAR:APCO25 0.2

sets the roll-off factor to 0.2 for filter type APCO25.

Manual operation: See "[Roll Off Factor / B xT](#)" on page 49

[:SOURce<hw>]:BB:BTOoth:FILTer:PARAmeter:COsine <Cosine>

The command sets the roll-off factor for the Cosine filter type.

Parameters:

<Cosine> float
 Range: 0 to 1.0
 Increment: 0.01
 *RST: 0.1

Example: BB:BTO:FILT:PAR:COs 0.35
 sets the roll-off factor to 0.35 for filter type Cosine.

Manual operation: See "Roll Off Factor / B x T" on page 49

[:SOURce<hw>]:BB:BTOoth:FILTer:PARAmeter:FGAUss <FGauss>

The command sets the B x T for the Gauss filter type.

Parameters:

<FGauss> float
 Range: 0.15 to 2.5
 Increment: 0.01
 *RST: 0.5

Example: BB:BTO:FILT:PAR:FGA 0.5
 sets B x T to 0.5 for the Gauss filter type for the GFSK section of the packet.

Manual operation: See "Roll Off Factor / B x T" on page 49

[:SOURce<hw>]:BB:BTOoth:FILTer:PARAmeter:GAUss <Gauss>

The command sets the B x T for the Gauss filter type.

Parameters:

<Gauss> float
 Range: 0.15 to 2.5
 Increment: 0.01
 *RST: 0.5

Example: BB:BTO:FILT:PAR:GAUS 0.5
 sets B x T to 0.5 for the Gauss filter type for $\pi/4$ DQPSK or 8DPSK sections.

Manual operation: See "Roll Off Factor / B x T" on page 49

[:SOURce<hw>]:BB:BTOoth:FILTer:PARAmeter:LPASs <LPass>

The command sets the cut off frequency factor for a lowpass filter (ACP Opt.).

Parameters:

<LPass> float
 Range: 0.05 to 2.00
 Increment: 0.01
 *RST: 0.50

Example:

BB:BTO:FILT:PAR:LPAS 1
 sets the cut off frequency factor for a lowpass filter

Manual operation: See ["Cut Off Frequency Factor"](#) on page 50

[[:SOURce<hw>]:BB:BTOoth:FILT:PAR:PGAuss <PGauss>

The command sets the B x T for the Pure Gauss filter type.

Parameters:

<PGauss> float
 Range: 0.15 to 2.5
 *RST: 0.5
 Default unit: 0.01

Example:

BB:BTO:FILT:PAR:PGA 0.5
 sets B x T to 0.5 for the Pure Gauss filter type.

Manual operation: See ["Roll Off Factor / B x T"](#) on page 49

[[:SOURce<hw>]:BB:BTOoth:FILT:PAR:RCOSine <RCosine>

The command sets the roll-off factor for the Root Cosine filter type.

Parameters:

<RCosine> float
 Range: 0 to 1.0
 Increment: 0.01
 *RST: 0.4

Example:

BB:BTO:FILT:PAR:RCOS 0.22
 sets the roll-off factor to 0.22 for filter type Root Cosine.

Manual operation: See ["Roll Off Factor / B x T"](#) on page 49

[[:SOURce<hw>]:BB:BTOoth:FILT:PAR:SPHase <SPhase>

The command sets the B x T for the Split Phase filter type.

Parameters:

<SPhase> float
 Range: 0.15 to 2.5
 Increment: 0.01
 *RST: 2

Example: `BB:BTO:FILT:PAR:SPH 0.5`
sets B x T to 0.5 for the Split Phase filter type.

Manual operation: See ["Roll Off Factor / B x T"](#) on page 49

[:SOURce<hw>]:BB:BTOoth:SRATe:VARIation <Variation>

The command enters the symbol rate.

Parameters:

<Variation> float
Range: 400 to 15E6
Increment: 1 Hz
*RST: 1E6
Default unit: Hz

Example: `BB:BTO:SRAT:VAR 1`
sets the symbol rate variation to 1 MHz.

Manual operation: See ["Symbol Rate Variation"](#) on page 51

4.3 Trigger Settings



The trigger settings are available for R&S SMx and R&S AMU instruments only.

EXTeRnal<ch>

The numeric suffix to EXTeRnal<ch> distinguishes between the external trigger via the TRIGGER 1 (suffix 1) and TRIGGER 2 (suffix 2) connector.

[:SOURce<hw>]:BB:BTOoth:TRIGger:ARM:EXECute.....	76
[:SOURce<hw>]:BB:BTOoth:TRIGger:EXECute.....	77
[:SOURce<hw>]:BB:BTOoth:TRIGger[:EXTeRnal]:SYNChronize:OUTPut.....	77
[:SOURce<hw>]:BB:BTOoth:TRIGger:OBASeband:DELay.....	78
[:SOURce<hw>]:BB:BTOoth:TRIGger:OBASeband:INHibit.....	78
[:SOURce<hw>]:BB:BTOoth:TRIGger:RMODE.....	78
[:SOURce<hw>]:BB:BTOoth:TRIGger:SLENgth.....	79
[:SOURce<hw>]:BB:BTOoth:TRIGger:SLUNit.....	79
[:SOURce<hw>]:BB:BTOoth:TRIGger:SOURce.....	80
[:SOURce<hw>]:BB:BTOoth:TRIGger[:EXTeRnal<ch>]:DELay.....	80
[:SOURce<hw>]:BB:BTOoth:TRIGger[:EXTeRnal<ch>]:INHibit.....	81
[:SOURce<hw>]:BB:BTOoth[:TRIGger]:SEQUence.....	81

[:SOURce<hw>]:BB:BTOoth:TRIGger:ARM:EXECute

The command stops signal generation for trigger modes "Armed Auto" and "Armed Retrigger". A subsequent internal or external trigger event restart signal generation.

Example: `BB:BTO:TRIG:ARM:EXEC`
stops signal generation for trigger modes "Armed Auto "and "Armed Retrigger".

Usage: Event

Manual operation: See ["Arm"](#) on page 19

[SOURce<hw>]:BB:BTOoth:TRIGger:EXECute

The command executes a trigger. The internal trigger source must be selected using the command `SOUR:BB:BTO:TRIG:SOUR INT` and a trigger mode other than "AUTO" must be selected using the command `SOUR:BB:BTO:TRIG:SEQ`.

Example: `BB:BTO:TRIG:SOUR INT`
sets internal triggering.
`BB:BTO:SEQ RETR`
sets Retrigger mode, i.e. every trigger event causes signal generation to restart.
`BB:BTO:TRIG:EXEC`
executes a trigger.

Usage: Event

Manual operation: See ["Execute Trigger"](#) on page 19

[SOURce<hw>]:BB:BTOoth:TRIGger[:EXTErnal]:SYNChronize:OUTPut <Output> (enabled for "Trigger Source" External)

Enables/disables output of the signal synchronous to the external trigger event.

For R&S SMBV instruments:

See also ["Sync. Output to External Trigger"](#) on page 56 for a detailed description of the applications of this setting.

Parameters:

<Output> 0 | 1 | OFF | ON

ON

The signal calculation starts simultaneously with the external trigger event but because of the instrument's processing time the first samples are cut off and no signal is outputted. After elapsing of the internal processing time, the output signal is synchronous to the trigger event.

OFF

The signal output begins after elapsing of the processing time and starts with sample 0, i.e. the complete signal is outputted.

*RST: ON

Example: `BB:BTO:TRIG:SOUR EXT`
sets external triggering.
`BB:BTO:TRIG:EXT:SYNC:OUTP ON`
enables synchronous output to external trigger

Manual operation: See ["Sync. Output to External Trigger"](#) on page 56

[[:SOURce<hw>]:BB:BTOoth:TRIGger:OBASeband:DELay <Delay>

The command specifies the trigger delay (expressed as a number of samples) for triggering by the trigger signal from the second path.

Parameters:

<Delay> float
Range: 0 samples to (1<<16)-1u samples
Increment: 0.01 samples
*RST: 0 samples

Example: `BB:BTO:TRIG:SOUR OBAS`
sets for path A the internal trigger executed by the trigger signal from the second path (path B).
`BB:BTO:TRIG:OBAS:DEL 50`
sets a delay of 50 symbols for the trigger.

Manual operation: See ["Trigger Delay"](#) on page 58

[[:SOURce<hw>]:BB:BTOoth:TRIGger:OBASeband:INHibit <Inhibit>

The command specifies the number of samples by which a restart is to be inhibited following a trigger event. This command applies only for triggering by the second path.

Parameters:

<Inhibit> integer
Range: 0 samples to (1<<26)-1u samples
Increment: 1 sample
*RST: 0 samples

Example: `BB:BTO:TRIG:SOUR OBAS`
sets for path A the internal trigger executed by the trigger signal from the second path (path B).
`BB:BTO:TRIG:OBAS:INH 200`
sets a restart inhibit for 200 samples following a trigger event.

Manual operation: See ["Trigger Inhibit"](#) on page 58

[[:SOURce<hw>]:BB:BTOoth:TRIGger:RMODE <RMode>

The command queries the current status of signal generation for all trigger modes with Bluetooth modulation on.

Parameters:

<RMode>

RUN | STOP

RUN

the signal is generated. A trigger event occurred in the triggered mode.

STOP

the signal is not generated. A trigger event did not occur in the triggered modes, or signal generation was stopped by the command :BB:BTO:TRIG:ARM:EXECute (armed trigger modes only).

Example:

BB:BTO:SEQ ARET

selects the Armed_Retrigger mode.

BB:BTO:TRIG:RMOD?

queries the current status of signal generation.

Response: RUN

the signal is generated, an external trigger was executed.

Manual operation: See ["Running/Stopped"](#) on page 55

[:SOURce<hw>]:BB:BTOoth:TRIGger:SENGth <SLength>

The command defines the length of the signal sequence to be output in the Single trigger mode (SOUR:BB:BTO:SEQ SING). The unit is defined with command SOUR:BB:BTO:TRIG:SLUNit.

Parameters:

<SLength>

float

Range: 1 to 7000

*RST: 1 sequence length

Example:

BB:BTO:SEQ SING

sets trigger mode Single.

BB:BTO:TRIG:SLUN FRAM

sets unit frames for the entry of sequence length.

BB:BTO:TRIG:SENG 200

sets a sequence length of 200 frames.

Manual operation: See ["Signal Duration"](#) on page 55

[:SOURce<hw>]:BB:BTOoth:TRIGger:SLUNit <SIUnit>

The command defines the unit for the entry of the length of the signal sequence (SOUR:BB:BTO:TRIG:SENG) to be output in the "Single" trigger mode (SOUR:BB:BTO:SEQ SING).

Parameters:

<SIUnit>

FRAME | SEQUENCE | EVENT

*RST: SEQUENCE

Example: BB:BTO:SEQ SING
sets trigger mode Single.
BB:BTO:TRIG:SLUN FRAM
sets unit frames for the entry of sequence length.
BB:BTO:TRIG:SLen 2
sets a sequence length of 2 frames. The current frame will be output twice after the next trigger event.

Manual operation: See ["Signal Duration Unit"](#) on page 55

[[:SOURce<hw>]:BB:BTOoth:TRIGger:SOURce <Source>

Selects the trigger source.

Parameters:

<Source> INTERNAL|OBASeband|BEXternal|EXTernal
INTERNAL
manual trigger or *TRG.
EXTernal | BEXternal
trigger signal on the TRIGGER 1/2 connector.
OBASeband
trigger signal from the other path
*RST: INTERNAL

Example: SOURce1:BB:BTOoth:TRIGger:SOURce EXTernal
sets external triggering via the TRIGGER 1 connector.

Manual operation: See ["Trigger Source"](#) on page 56

[[:SOURce<hw>]:BB:BTOoth:TRIGger[:EXTernal<ch>]:DELay <Delay>

The command specifies the trigger delay (expressed as a number of samples) for external triggering.

The numeric suffix to EXTernal distinguishes between the external trigger via the TRIGGER 1 (suffix 1) and TRIGGER 2 (suffix 2) connector.

Parameters:

<Delay> float
Range: 0 samples to 65535 samples
Increment: 0.01 samples
*RST: 0 samples

Example: BB:BTO:TRIG:SOUR EXT
sets an external trigger via the TRIGGER 1 connector.
BB:BTO:TRIG:EXT:DEL 50
sets a delay of 50 symbols for the trigger.

Manual operation: See ["Trigger Delay"](#) on page 58

[[:SOURce<hw>]:BB:BTOoth:TRIGger[:EXTeRnal<ch>]:INHibit <Inhibit>

The command specifies the number of samples by which a restart is to be inhibited following a trigger event. This command applies only in the case of external triggering. The numeric suffix to EXTeRnal distinguishes between the external trigger via the TRIGGER 1 (suffix 1) and TRIGGER 2 (suffix 2) connector.

Parameters:

<Inhibit> float
 Range: 0 samples to (1<<26)-1u samples
 Increment: 1 sample
 *RST: 0 samples

Example:

BB:BTO:TRIG:SOUR EXT
 selects an external trigger via the TRIGGER 1 connector.
 BB:BTO:TRIG:EXT:INH 200
 sets a restart inhibit for 200 samples following a trigger event.

Manual operation: See ["Trigger Inhibit"](#) on page 58

[[:SOURce<hw>]:BB:BTOoth[:TRIGger]:SEqueNce <Sequence>

The command selects the trigger mode.

Parameters:

<Sequence> AUTO | RETRigger | AAUTo | ARETRigger | SINGLE

AUTO

The modulation signal is generated continuously.

RETRigger

The modulation signal is generated continuously. A trigger event (internal or external) causes a restart.

AAUTo

The modulation signal is generated only when a trigger event occurs. After the trigger event the signal is generated continuously, signal generation is stopped with command SOUR:BB:BTO:TRIG:ARM:EXEC and started again when a trigger event occurs.

ARETRigger

The modulation signal is generated only when a trigger event occurs. The device automatically toggles to RETRIG mode. Every subsequent trigger event causes a restart.

Signal generation is stopped with command

SOUR:BB:BTO:TRIG:ARM:EXEC and started again when a trigger event occurs.

SINGLE

The modulation signal is generated only when a trigger) event occurs. After the trigger event, the signal is generated once to the set sequence length (SOUR:BB:BTO:TRIG:SLEN). Every subsequent trigger event causes a restart.

*RST: AUTO

Example:

BB:BTO:SEQ AAUT

sets the "Armed_auto" trigger mode; the device waits for the first trigger (e.g. with *TRG) and then generates the signal continuously.

Manual operation: See ["Trigger Mode"](#) on page 54

4.4 Marker Settings

This section lists the remote control commands, necessary to configure the markers.



The marker delay settings are available for R&S SMx and R&S AMU instruments only.

OUTPut<ch>

The numeric suffix to OUTPut distinguishes between the available markers.

Only two markers are available for the R&S SMBV, i.e. the allowed values for the suffix are 1 or 2.

[:SOURce<hw>]:BB:BT00th:TRIGger:OUTPut:DELay:FIXed	82
[:SOURce<hw>]:BB:BT00th:TRIGger:OUTPut<ch>:DELay	83
[:SOURce<hw>]:BB:BT00th:TRIGger:OUTPut<ch>:DELay:MAXimum?	83
[:SOURce<hw>]:BB:BT00th:TRIGger:OUTPut<ch>:DELay:MINimum?	83
[:SOURce<hw>]:BB:BT00th:TRIGger:OUTPut<ch>:MODE	84
[:SOURce<hw>]:BB:BT00th:TRIGger:OUTPut<ch>:ONTime	85
[:SOURce<hw>]:BB:BT00th:TRIGger:OUTPut<ch>:OFFTime	85
[:SOURce<hw>]:BB:BT00th:TRIGger:OUTPut<ch>:PATTern	85
[:SOURce<hw>]:BB:BT00th:TRIGger:OUTPut<ch>:PULSe:DIVider	85
[:SOURce<hw>]:BB:BT00th:TRIGger:OUTPut<ch>:PULSe:FREQuency?	86

[\[:SOURce<hw>\]:BB:BT00th:TRIGger:OUTPut:DELay:FIXed](#) <Fixed>

The command restricts the marker delay setting range to the current range. In this range the delay can be set without restarting the marker and signal. If a delay is entered in setting ON but is outside this range, the maximum possible delay is set and an error message is generated.

The numeric suffix in OUTPut has no significance for this command, since the setting always affects every marker.

Parameters:

<Fixed> 0 | 1 | OFF | ON
 *RST: OFF

Example:

BB:BTO:TRIG:OUTP:DEL:FIX ON

restricts the marker signal delay setting range to the current range.

Manual operation: See ["Fix marker delay to current range"](#) on page 60

[[:SOURce<hw>]:BB:BTOoth:TRIGger:OUTPut<ch>:DELay <Delay>

The command defines the delay between the signal on the marker outputs and the start of the signal, expressed in terms of samples. Command

BB:BTO:TRIGger:OUTPut:DELay:FIXed can be used to restrict the range of values to the dynamic range, i.e. the range within which a delay of the marker signals can be set without restarting the marker and signal.

Parameters:

<Delay> float
 Range: 0 samples to ((1<<24)-1u) samples
 Increment: 0.001 samples
 *RST: 0 samples

Example: BB:BTO:TRIG:OUTP2:DEL 1600
 sets a delay of 1600 samples for the corresponding marker signal.

Manual operation: See "[Marker x Delay](#)" on page 60

[[:SOURce<hw>]:BB:BTOoth:TRIGger:OUTPut<ch>:DELay:MAXimum?

The command queries the maximum marker delay for setting :BB:BTO:TRIG:OUTP:DEL:FIX ON.

Return values:

<Maximum> float

Example: BB:BTO:TRIG:OUTP:DEL:FIX ON
 restricts the marker signal delay setting range to the dynamic range.
 BB:BTO:TRIG:OUTP:DEL:MAX?
 queries the maximum of the dynamic range.
 Response: 2000
 the maximum for the marker delay setting is 2000 samples.

Usage: Query only

Manual operation: See "[Current Range without Recalculation](#)" on page 60

[[:SOURce<hw>]:BB:BTOoth:TRIGger:OUTPut<ch>:DELay:MINimum?

The command queries the minimum marker delay for setting :BB:BTO:TRIGger:OUTPut:DELay:FIXed ON.

Return values:

<Minimum> float

- Example:** BB:BTO:TRIG:OUTP:DEL:FIX ON
restricts the marker signal delay setting range to the dynamic range.
BB:BTO:TRIG:OUTP:DEL:MIN?
queries the minimum of the dynamic range.
Response: 0
the minimum for the marker delay setting is 0 samples.
- Usage:** Query only
- Manual operation:** See ["Current Range without Recalculation"](#) on page 60

[:SOURce<hw>]:BB:BTOoth:TRIGger:OUTPut<ch>:MODE <Mode>

The command defines the signal for the selected marker output.

Parameters:

<Mode>

REStart | START | ACTive | PULSe | PATtern | RATio | TRIGger

REStart

A marker signal is generated at the start of each signal sequence.

START

A marker signal is generated at the start of each frame

ACTive

The marker masks the active part of the frame. At the start of each burst, the marker signal changes to high. It changes back to low after the end of each burst.

PULSe

A regular marker signal is generated. The clock frequency is defined by entering a divider. The frequency is derived by dividing the symbol rate by the divider. The input box for divider opens when Pulse is selected, and the resulting pulse frequency is displayed below.

PATtern

A marker signal that is defined by a bit pattern is generated. The pattern has a maximum length of 32 bits and is defined in an input field which opens when pattern is selected.

RATio

A regular marker signal corresponding to the Time Off / Time On specifications in the commands

SOURce:BB:BTO:TRIGger:OUTPut:OFFTime and

SOURce:BB:BTO:TRIGger:OUTPut:ONTime is generated.

TRIGger

A received internal or external trigger signal is output at the marker connector.

*RST: REStart

Example: `BB:BTO:TRIG:OUTP:MODE REST`
selects the marker generation at the start of each signal sequence

Manual operation: See "[Marker Mode](#)" on page 59

[SOURce<hw>]:BB:BTOoth:TRIGger:OUTPut<ch>:ONTime <OnTime>
[SOURce<hw>]:BB:BTOoth:TRIGger:OUTPut<ch>:OFFTime <OffTime>

The command sets the number of samples in a period (ON time + OFF time) during which the marker signal in setting `SOURce:BB:BTO:TRIGger:OUTPut:MODE RATio` on the marker outputs is OFF.

Parameters:

<OffTime> float
Range: 1 sample to (1<<24)-1u samples
Increment: 1 sample
*RST: 1 sample

Example: `BB:BTO:TRIG:OUTP:OFFT 2000`
sets an OFF time of 2000 samples for the corresponding marker signal.

Manual operation: See "[Marker Mode](#)" on page 59

[SOURce<hw>]:BB:BTOoth:TRIGger:OUTPut<ch>:PATtern <Pattern>

The command selects the data for a pattern.

Parameters:

<Pattern> integer

Example: `BB:BTO:TRIG:OUTP:MODE PATT`
sets the marker mode for the corresponding marker signal
`BB:BTO:TRIG:OUTP:PATT #B010101, 6`
sets the pattern

Manual operation: See "[Marker Mode](#)" on page 59

[SOURce<hw>]:BB:BTOoth:TRIGger:OUTPut<ch>:PULSe:DIVider <Divider>

The clock frequency is defined by entering a divider. The frequency is derived by dividing the symbol rate by the divider. The input box for divider opens when Pulse is selected, and the resulting pulse frequency is displayed below it.

Parameters:

<Divider> float
Range: 2 to 1024
Increment: 1
*RST: 2

Example: BB:BTO:TRIG:OUTP:MODE:PULS
selects marker mode Pulse
BB:BTO:TRIG:OUTP:PULS:DIV 2
sets a divider for the clock frequency.

Manual operation: See "Marker Mode" on page 59

[:SOURce<hw>]:BB:BTOoth:TRIGger:OUTPut<ch>:PULSe:FREQUENCY?

The clock frequency is defined by entering a divider. The frequency is derived by dividing the symbol rate by the divider. The input box for divider opens when Pulse is selected, and the resulting pulse frequency is displayed below it.

Return values:

<Frequency> float

Example: BB:BTO:TRIG:OUTP:MODE:PULS
selects marker mode Pulse
BB:BTO:TRIG:OUTP:PULS:DIV 2
sets a divider for the clock frequency
BB:BTO:TRIG:OUTP:PULS:FREQ
displays resulting pulse frequency

Usage: Query only

Manual operation: See "Marker Mode" on page 59

4.5 Clock Settings

This section lists the remote control commands, necessary to configure the clock.



The clock settings are available for R&S SMx and R&S AMU instruments only.

[:SOURce<hw>]:BB:BTOoth:CLOCK:MODE.....	86
[:SOURce<hw>]:BB:BTOoth:CLOCK:MULTiplier.....	87
[:SOURce<hw>]:BB:BTOoth:CLOCK:SOURce.....	87
[:SOURce<hw>]:BB:BTOoth:CLOCK:SYNChronization:EXECute.....	88
[:SOURce<hw>]:BB:BTOoth:CLOCK:SYNChronization:MODE.....	88

[:SOURce<hw>]:BB:BTOoth:CLOCK:MODE <Mode>

The command enters the type of externally supplied clock. When MSAMple is used, a multiple of the clock is supplied and the clock is derived internally from this. The multiplier is entered with the command [:SOURce<hw>]:BB:BTOoth:CLOCK:MULTiplier.

For two-path instruments, the only numerical suffix allowed for SOURce is 1, since the external clock source is permanently allocated to path A.

Parameters:

<Mode> SAMPLE | MSAMPLE
 *RST: SAMP

Example:

SOURce1:BB:BTO:CLOCK:MODE MSAMPLE
 sets the type of externally supplied clock.

Manual operation: See "Clock Mode" on page 61

[:SOURce<hw>]:BB:BTOoth:CLOCK:MULTIplier <Multiplier>

Specifies the multiplication factor for clock type multiple (:BB:BTO:CLOCK:MODE MSAMPLE) in the case of an external clock source.

For two-path instruments, the only numerical suffix allowed for SOURce is 1, since the external clock source is permanently allocated to path A.

Parameters:

<Multiplier> float
 Range: 1 to 64
 Increment: 1
 *RST: 4

Example:

SOURce1:BB:BTO:CLOCK:SOURce EXT
 selects the external clock source.
 SOURce1:BB:BTO:CLOCK:MODE MSAMPLE
 sets the clock type
 SOURce1:BB:BTO:CLOCK:MULTIplier 12
 the multiplier for the external clock rate is 12.

Manual operation: See "Clock Multiplier" on page 62

[:SOURce<hw>]:BB:BTOoth:CLOCK:SOURce <Source>

The command selects the clock source.

For two-path instruments, selecting EXTERNAL is only possible for path A, since the external clock source is permanently allocated to path A. Selection AINTERNAL is only possible for path B.

Parameters:

<Source> INTERNAL | EXTERNAL | AINTERNAL

INTERNAL

The internal clock reference is used.

EXTERNAL

The external clock reference is supplied to the CLOCK connector. Commands :BB:BTO:CLOCK:MODE and :MULTIplier are used to enter the type of the external clock.

AINTERNAL

The clock source of path A is used for path B.

*RST: INTERNAL

Example: `BB:BTO:CLOC:SOUR EXT`
 selects the external clock source. The clock is supplied via the CLOCK connector.
`BB:BTO:CLOC:MODE MSAM`
 selects clock type "Multiple Sample", i.e. the supplied clock has a rate which is a multiple of the sample rate.
`BB:BTO:CLOC:MULT 12`
 the multiplier for the external clock rate is 12.

Manual operation: See ["Clock Source"](#) on page 61

[:SOURce<hw>]:BB:BTOoth:CLOCK:SYNChronization:EXECute

Performs automatically adjustment of the instrument's settings required for the synchronization mode, set with the command `BB:BTO:CLOC:SYNC:MODE`.

Example: `BB:BTO:CLOC:SYNC:MODE MAST`
 the instrument is configured to work as a master one.
`BB:BTO:CLOC:SYNC:EXEC`
 all synchronizations settings are adjusted accordingly.

Usage: Event

Manual operation: See ["Set Synchronization Settings"](#) on page 61

[:SOURce<hw>]:BB:BTOoth:CLOCK:SYNChronization:MODE <Mode>

Selects the synchronization mode.

This parameter is used to enable generation of very precise synchronous signal of several connected R&S SMBVs.

Note: If several instruments are connected, the connecting cables from the master instrument to the slave one and between each two consecutive slave instruments must have the same length and type. Avoid unnecessary cable length and branching points.

Parameters:

<Mode> NONE | MASTer | SLAVe

NONE

The instrument is working in stand-alone mode.

MASTer

The instrument provides all connected instrument with its synchronization (including the trigger signal) and reference clock signal.

SLAVe

The instrument receives the synchronization and reference clock signal from another instrument working in a master mode.

*RST: NONE

Example: `BB:BTO:CLOC:SYNC:MODE MAST`
 the instrument is configured to work as a master one.

Manual operation: See ["Sync. Mode"](#) on page 60

4.6 Power Ramping

[:SOURce<hw>]:BB:BT0oth:PRAMping:FOFFset	89
[:SOURce<hw>]:BB:BT0oth:PRAMping:RFUNction	89
[:SOURce<hw>]:BB:BT0oth:PRAMping:ROFFset	89
[:SOURce<hw>]:BB:BT0oth:PRAMping:RTIME	90

`[:SOURce<hw>]:BB:BT0oth:PRAMping:FOFFset <FOffset>`

The command sets the offset of the falling edge of the envelope at the end of a burst. A positive value introduces a guard period after the end of the packet and negative value moves the ramp into the end part of the transmitted packet.

Parameters:

`<FOffset>` integer
 Range: -32 to 32
 Increment: 1
 *RST: 0
 Default unit: symbols

Example: `BB:BT0:PRAM:FOFF 8.0`
 sets the offset in the falling edge of the envelope to 8.0 symbols.

Manual operation: See ["Fall Offset"](#) on page 53

`[:SOURce<hw>]:BB:BT0oth:PRAMping:RFUNction <RFunction>`

The command selects the form of the transmitted power, i.e. the shape of the rising and falling edges during power ramp control.

Parameters:

`<RFunction>` LINear | COSine
 *RST: COSine

Example: `BB:BT0:PRAM:RFUN LIN`
 sets linear shape for the rising and falling edges during power ramp control.

Manual operation: See ["Ramp Function"](#) on page 52

`[:SOURce<hw>]:BB:BT0oth:PRAMping:ROFFset <ROffset>`

The command sets the offset in the rising edge of the envelope at the start of a burst. A positive value moves the ramp into beginning of a transmitted packet and a negative value introduce an additional guard period after the end of the packet.

Parameters:

<ROffset> integer
 Range: -32 symbols to 32 symbols
 Increment: 1 symbol
 *RST: 0 symbols

Example:

BB:BTO:PRAM:ROFF 8.0
 sets the offset in the rising edge of the envelope to 8.0 symbols.

Manual operation: See ["Rise Offset"](#) on page 53

[SOURce<hw>]:BB:BTOoth:PRAMping:RTIME <RTime>

The command sets the power ramping rise time and fall time for a burst.

Parameters:

<RTime> integer
 Range: 1 symbol to 32 symbols
 Increment: 1 symbol
 *RST: 1 symbol

Example:

BB:BTO:PRAM:TIME 2.0
 sets the power ramping rise time and fall time for a burst to 2 samples.

Manual operation: See ["Ramp Time"](#) on page 53

4.7 Packet Configuration Setting

[:SOURce<hw>]:BB:BTOoth:PCONfiguration:ACKnowledgement.....	91
[:SOURce<hw>]:BB:BTOoth:PCONfiguration:BDALap.....	91
[:SOURce<hw>]:BB:BTOoth:PCONfiguration:BDANap.....	91
[:SOURce<hw>]:BB:BTOoth:PCONfiguration:BDUap.....	92
[:SOURce<hw>]:BB:BTOoth:PCONfiguration:CODEvice.....	92
[:SOURce<hw>]:BB:BTOoth:PCONfiguration:DATA.....	92
[:SOURce<hw>]:BB:BTOoth:PCONfiguration:DATA:DPATtern.....	93
[:SOURce<hw>]:BB:BTOoth:PCONfiguration:DATA:DSELection.....	93
[:SOURce<hw>]:BB:BTOoth:PCONfiguration:DATA:VDPAttern.....	93
[:SOURce<hw>]:BB:BTOoth:PCONfiguration:DATA:VDSElection.....	94
[:SOURce<hw>]:BB:BTOoth:PCONfiguration:DLEngth.....	94
[:SOURce<hw>]:BB:BTOoth:PCONfiguration:DSFPacket.....	94
[:SOURce<hw>]:BB:BTOoth:PCONfiguration:DWHitening.....	95
[:SOURce<hw>]:BB:BTOoth:PCONfiguration:EIRPacketfollows.....	95
[:SOURce<hw>]:BB:BTOoth:PCONfiguration:HFControl.....	96
[:SOURce<hw>]:BB:BTOoth:PCONfiguration:LTADdress.....	96
[:SOURce<hw>]:BB:BTOoth:PCONfiguration:PFControl.....	96
[:SOURce<hw>]:BB:BTOoth:PCONfiguration:PLENght.....	97

[:SOURce<hw>]:BB:BTOoth:PCONfiguration:SNValue.....	97
[:SOURce<hw>]:BB:BTOoth:PCONfiguration:SRMode.....	98
[:SOURce<hw>]:BB:BTOoth:PCONfiguration:VDATa.....	98

`[:SOURce<hw>]:BB:BTOoth:PCONfiguration:ACKNowledge` `<Acknowledgement>`

This command sets the ARQN bit of the packet header..

Parameters:

`<Acknowledgement>` NAK | ACK

NAK

Request to retransmit the previous payload.

ACK

Previous payload has been received successfully.

*RST: ACK

Example:

`BB:BTO:PTYP DH1`

selects the packet type DH1.

`BB:BTO:PCON:DSFP PED`

enable packet editor under data source for packet

`BB:BTO:PCON:ACKN ACK`

sets positive acknowledgement

Manual operation: See "[Acknowledgment](#)" on page 23

`[:SOURce<hw>]:BB:BTOoth:PCONfiguration:BDALap <BdaLap>`

Enters the lower address part of Bluetooth Device Address. The length of LAP is 24 bits or 6 hexadecimal figures.

Parameters:

`<BdaLap>` integer

Range: #H000000 to #FFFFFFF

*RST: 80

Example:

`BB:BTO:PCON:BDAL #H000000, 24`

sets the lower address part

Manual operation: See "[Bluetooth Device Address \(BD_ADDR\)](#)" on page 22

`[:SOURce<hw>]:BB:BTOoth:PCONfiguration:BDANap <BdaNap>`

Enters the non-significant address part of Bluetooth Device Address. The length of NAP is 16 bits or 4 hexadecimal figures.

Parameters:

`<BdaNap>` integer

Range: #H0000 to #FFFF

*RST: ABCD

Example: BB:BTO:PCON:BDAN #H0000, 16
sets the non-significant address part

Manual operation: See ["Bluetooth Device Address \(BD_ADDR\)"](#) on page 22

[:SOURce<hw>]:BB:BTOoth:PCONfiguration:BDAUap <BdaUap>

Enters the upper address part of Bluetooth Device Address. The length of UAP is 8 bits or 2 hexadecimal figures.

Parameters:

<BdaUap> integer
Range: #H00 to #HFF
Increment: 1
*RST: 48

Example: BB:BTO:PCON:BDAN #H00, 8
sets the non-significant address part

Manual operation: See ["Bluetooth Device Address \(BD_ADDR\)"](#) on page 22

[:SOURce<hw>]:BB:BTOoth:PCONfiguration:CODevice <CoDevice>

A parameter received during the device discovery procedure, indicates the type of device and which types of service that are supported.

Parameters:

<CoDevice> integer
Range: #H000000 to #HFFFFFFF

Example: BB:BTO:PTYP FHS
sets the packet type
BB:BTO:PCON:DSFP PED
enable packet editor under data source for packet
BB:BTO:PCON:COD '020104'
sets the class of device

Manual operation: See ["Class of Device"](#) on page 25

[:SOURce<hw>]:BB:BTOoth:PCONfiguration:DATA <Data>

Selects the data source used for the payload.

Parameters:

<Data> ALL0 | ALL1 | PATTeRn | PN09 | PN11 | PN15 | PN16 | PN20 |
PN21 | PN23 | DLISt
*RST: PN09

Example: BB:BTO:PTYP FHS
sets the packet type
BB:BTO:PCON:DSFP PED
enable packet editor under data source for packet
BB:BTO:PCON:DATA ALL1
sets the data type.

Manual operation: See ["Data List Management..."](#) on page 17

[[:SOURce<hw>]:BB:BTOoth:PCONfiguration:DATA:DPATtern <DPattern>

The command selects the data for a pattern.

Parameters:

<DPattern> integer
Increment: 1
*RST: 0

Example: BB:BTO:PCON:DATA PATT
sets the data type.
BB:BTO:PCON:DATA:DPAT #B010101, 6
selects the data for a pattern

Manual operation: See ["Data Source"](#) on page 23

[[:SOURce<hw>]:BB:BTOoth:PCONfiguration:DATA:DSElection <DSelection>

The command selects data list file.

Parameters:

<DSelection> string
Increment: 1

Example: BB:BTO:PCON:DATA DLIS
selects the data type.
BB:BTO:PCON:DSEL bluetooth_1
selects the file for the data.

Manual operation: See ["Data List Management..."](#) on page 17

[[:SOURce<hw>]:BB:BTOoth:PCONfiguration:DATA:VDPattern <VdPattern>

The command selects the bit pattern for the voice data.

Parameters:

<VdPattern> integer
Increment: 1
*RST: 0

Example: BB:BTO:PCON:DATA:PATT
selects the data type.
BB:BTO:PCON:DATA:VDPA #B010101, 6
selects the bit pattern for the voice data.

Manual operation: See ["Data Source \(Voice Field\)"](#) on page 25

[:SOURce<hw>]:BB:BTOoth:PCONfiguration:DATA:VDSElection <VdSelection>

The command selects the data list for voice data.

Parameters:

<VdSelection> string
Increment: 1

Example: BB:BTO:PCON:VDAT DLIS
selects the data type.
BB:BTO:PCON:VDSE bluetooth_1
selects the file for the data.

Manual operation: See ["Data Source \(Voice Field\)"](#) on page 25

[:SOURce<hw>]:BB:BTOoth:PCONfiguration:DLENgth <DLength>

The command enters the payload data length in bytes.

Parameters:

<DLength> float
Range: 0 to 1021
Increment: 1
*RST: 0

Example: BB:BTO:PTYP DH1
sets the packet type.
BB:BTO:PCON:DSFP PED
enable packet editor under data source for packet
BB:BTO:PCON:DLEN 25
sets the data length.

Manual operation: See ["Data Length"](#) on page 24

[:SOURce<hw>]:BB:BTOoth:PCONfiguration:DSFPacket <DsfPacket>

The command selects the data source for the selected packet type.

Parameters:

<DsfPacket>

PEdit | ADATa

PED

Enables Packet Editor. All packet fields can be configured individually.

ADAT

Fills the generated packets with the selected data source. Useful if predefined data contents are loaded with a data list file or the data contents of the packet are not of interest.

*RST: PEdit

Example:

BB:BTO:PCON:DSFP PED

enables packet editor under data source for packet.

Manual operation: See ["Data Source for Packet"](#) on page 21

[:SOURce<hw>]:BB:BTOoth:PCONfiguration:DWhitening <DWhitening>

The command activates or deactivates the Data Whitening.

Parameters:

<DWhitening>

0 | 1 | OFF | ON

*RST: OFF

Example:

BB:BTO:PCON:DWH ON

activates data whitening.

Manual operation: See ["Data Whitening"](#) on page 22

[:SOURce<hw>]:BB:BTOoth:PCONfiguration:EIRPacketfollows <EirPacketFollow>

The command indicates that an extended inquiry response packet may follow.

Parameters:

<EirPacketFollow>

YES | NO

YES

Indicates that EIR packet follows.

NO

Indicates that EIR packet does not follow.

*RST: NO

Example:

BB:BTO:PCON:PTYP FHS

sets the packet type.

BB:BTO:PCON:DSFP PED

enable packet editor under data source for packet

BB:BTO:PCON:EIRP YES

the EIR packet follows.

Manual operation: See ["EIR packet follows"](#) on page 24

[[:SOURce<hw>]:BB:BT0oth:PCONfiguration:HFControl <HfControl>

The command sets the FLOW bit in the header. This bit indicates start or stop of transmission of packets over the ACL logical transport.

Parameters:

<HfControl> GO | STOP

GO

Allows the other devices to transmit new data.

STOP

Stops the other devices from transmitting data temporarily.

*RST: GO

Example:

BB: BTO: PCON: PTYP DH1

sets the packet type.

BB: BTO: PCON: DSFP PED

enable packet editor under data source for packet.

BB: BTO: PCON: HFC GO

allows the other devices to transmit new data.

Manual operation: See ["Flow Control \(Header\)"](#) on page 22

[[:SOURce<hw>]:BB:BT0oth:PCONfiguration:LTAddress <LtAddress>

The command enters the logical transport address for the header. Each slave active in a piconet is assigned a primary logical transport address (LT_ADDR). The all-zero LT_ADDR is reserved for broadcast messages.

Parameters:

<LtAddress> integer

Range: 0 to 7

Increment: 1

*RST: 0

Example:

BB: BTO: PCON: PTYP DH1

sets the packet type.

BB: BTO: PCON: DSFP PED

enable packet editor under data source for packet

BB: BTO: PCON: LTAD 0

sets the logical transport address equal zero.

Manual operation: See ["Logical Transport Address"](#) on page 22

[[:SOURce<hw>]:BB:BT0oth:PCONfiguration:PFControl <PfControl>

The command sets the FLOW bit in the payload (flow control per logical link).

Parameters:

<PfControl>

GO | STOP

GO

Indicates the start of transmission of ACL packets after a new connection has been established.

STOP

Indicates the stop of transmission of ACL packets before an additional amount of payload data is sent.

*RST: GO

Example:

```
BB:BTO:PCON:PTYP DH1
```

sets the packet type.

```
BB:BTO:PCON:DSFP PED
```

enable packet editor under data source for packet

```
BB:BTO:PCON:PFC GO
```

allows the flow per logical link.

Manual operation: See ["Flow Control \(Payload\)"](#) on page 24

[:SOURce<hw>]:BB:BTOoth:PCONfiguration:PLENgtH <PLength>

The command enters the packet length in symbols.

Parameters:

<PLength>

float

Range: 1 to 2873

Increment: 1

*RST: 0

Example:

```
BB:BTO:PCON:DSFP ADAT
```

fills the all data under data source for packet.

```
BB:BTO:PCON:PLEN 1
```

sets the packet length.

Manual operation: See ["Packet Length"](#) on page 24

[:SOURce<hw>]:BB:BTOoth:PCONfiguration:SNSValue <SnSvalue>

The command sets the start value of the header SEQN bit. The SEQN bit is present in the header to filter out retransmissions in the destination. The signal generator is altering this bit automatically on consecutive frames, if a sequence length of at least 2 frames is set.

Parameters:

<SnSvalue>

integer

Range: 0 to 1

Increment: 1

*RST: 1

Example: BB:BTO:PCON:PTYP DH1
sets the packet type.
BB:BTO:PCON:DSFP PED
enables packet editor under data source for packet.
BB:BTO:PCON:SNSV ONE
sets the SEQN bit of the first CRC data packet at the start of a connection.

Manual operation: See ["SEQN Start Value"](#) on page 23

[[:SOURce<hw>]:BB:BT0oth:PCONfiguration:SRMode <SrMode>

The command indicates the interval between two consecutive page scan windows, determines the behavior of the paging device.

Parameters:

<SrMode> R0 | R1 | R2

R0
The scan interval is equal to the scan window T w page scan (continuous nscan) and maximal 1.28s.

R1
The scan interval is maximal 1.28s.

R2
The scan interval is maximal 2.56s.

*RST: R0

Example: BB:BTO:PCON:PTYP FHS
sets the packet type.
BB:BTO:PCON:DSFP PED
enables packet editor under data source for packet.
BB:BTO:PCON:SRM R0
sets the scan repetition mode.

Manual operation: See ["Scan Repetition Mode"](#) on page 24

[[:SOURce<hw>]:BB:BT0oth:PCONfiguration:VDATa <VData>

The command selects the data source for the voice field.

Parameters:

<VData> ALL0 | ALL1 | PATTeRn | PN09 | PN11 | PN15 | PN16 | PN20 | PN21 | PN23 | DLISt

*RST: PN09

Example: BB:BTO:PCON:VDAT ALL1
sets the voice data type.

Manual operation: See ["Data Source \(Voice Field\)"](#) on page 25

4.8 Dirty Transmitter Test

[:SOURce<hw>]:BB:BTOoth:DTTest:DTTState	99
[:SOURce<hw>]:BB:BTOoth:DTTest:FDDeviation	99
[:SOURce<hw>]:BB:BTOoth:DTTest:FDRate	100
[:SOURce<hw>]:BB:BTOoth:DTTest:SPHase	100
[:SOURce<hw>]:BB:BTOoth:DTTest:STDefault	100
[:SOURce<hw>]:BB:BTOoth:DTTest:TABLE	100
[:SOURce<hw>]:BB:BTOoth:DTTest:TABLE:LONG:SET<ch>:CFOffset	101
[:SOURce<hw>]:BB:BTOoth:DTTest:TABLE:LONG:SET<ch>:MINDex	101
[:SOURce<hw>]:BB:BTOoth:DTTest:TABLE:LONG:SET<ch>:STATE	102
[:SOURce<hw>]:BB:BTOoth:DTTest:TABLE:LONG:SET<ch>:STERror	102
[:SOURce<hw>]:BB:BTOoth:DTTest:TABLE:SHORT:SET<ch>:CFOffset	102
[:SOURce<hw>]:BB:BTOoth:DTTest:TABLE:SHORT:SET<ch>:STATE	103
[:SOURce<hw>]:BB:BTOoth:DTTest:TABLE:SHORT:SET<ch>:STERror	103

[\[:SOURce<hw>\]:BB:BTOoth:DTTest:DTTState](#) <DttState>

The command activates/deactivates the Dirty Transmitter Test. For Basic Rate packets, each set of parameters in the Dirty Transmitter Setting table below is used for a duration of 20 ms. After 20 ms, the following set is used, continuing with the first set after the sequence is completed.

For EDR packets, the parameter sets apply for 20 packets each.

Parameters:

<DttState> 0 | 1 | OFF | ON
 *RST: OFF

Example:

BB:BTO:DTT:DTTS ON
 activates the Dirty Transmitter Test.

Manual operation: See ["Dirty Transmitter Test"](#) on page 46

[\[:SOURce<hw>\]:BB:BTOoth:DTTest:FDDeviation](#) <FdDeviation>

The command enters a frequency drift rate.

A sine wave is used to drift the modulated Bluetooth signal around center frequency + carrier frequency offset. The maximum deviation reached during the drift equals the set frequency drift deviation.

Parameters:

<FdDeviation> float
 Range: -100 kHz to 100 kHz
 Increment: 1 KHz
 *RST: 25 kHz

Example:

BB:BTO:DTT:FDD 25
 enters frequency drift deviation.

Manual operation: See ["Frequency Drift Deviation \(+/-\)"](#) on page 47

[[:SOURce<hw>]:BB:BTOoth:DTTTest:FDRate <FdRate>

The command enters a frequency drift rate.

A sine wave is used to drift the modulated Bluetooth signal around center frequency + carrier frequency offset with the set frequency drift rate.

Parameters:

<FdRate> 0.3 KHz | 0.5 KHz | 1.6 KHz | 10 KHz
 Range: 0.3 kHz to 10 kHz
 *RST: 1.6 kHz

Example: BB:BTO:DTT:FDR 1.6
 enters frequency drift rate.

Manual operation: See ["Frequency Drift Rate"](#) on page 47

[[:SOURce<hw>]:BB:BTOoth:DTTTest:SPHase <SPHase>

The command enters a start phase.

The start phase of the sine wave used to drift the modulated Bluetooth signal around center frequency + carrier frequency offset is set here.

Parameters:

<SPHase> integer
 Range: 0 to 359
 Increment: 1
 *RST: 0
 Default unit: degree

Example: BB:BTO:DTT:SPH 0
 enters a start phase.

Manual operation: See ["Start Phase"](#) on page 47

[[:SOURce<hw>]:BB:BTOoth:DTTTest:STDefault

The command calls the default settings for the Dirty Transmitter Test.

Example: BB:BTO:DTT:STD
 calls the default settings.

Usage: Event

Manual operation: See ["Set to Default"](#) on page 46

[[:SOURce<hw>]:BB:BTOoth:DTTTest:TABLE <Table>

The command calls the table settings.

Parameters:

<Table> NOTable | SHORT | LONG

Example: BB:BTO:PTYP DH1
calls the default settings.
BB:BTO:DTT:TABL LONG
calls the default settings.

[:SOURce<hw>]:BB:BT0oth:DTTTest:TABLE:LONG:SET<ch>:CFOffset <CfOffset>

The command enters a carrier frequency offset.

The carrier frequency offset shows the deviation of the transmitted initial center frequency from carrier frequency.

Parameters:

<CfOffset> float
Range: -150 kHz to 150 kHz
Increment: 1 kHz

Example: BB:BTO:PTYP DH1
sets the packet type.
BB:BTO:DTT:TABL LONG
enters the table type
BB:BTO:DTT:TABL:LONG:SET2:CFOF 14
enters a carrier frequency offset.

Manual operation: See ["Carrier Frequency Offset kHz"](#) on page 47

[:SOURce<hw>]:BB:BT0oth:DTTTest:TABLE:LONG:SET<ch>:MINDex <MIndex>

The command enters the modulation index.

The modulation index specifies the frequency deviation..

The modulation index h is defined as

$$h = \frac{2\Delta f}{f_{symbol}}$$

where

f_{symbol} = "symbol rate", set with the command [\[:SOURce<hw>\]:BB:BT0oth:SRATe:VARiation](#)

Δf = "frequency deviation", set with the command [\[:SOURce<hw>\]:BB:BT0oth:MSETtings:FDEVIation](#)

According to the Bluetooth standard, the modulation index is allowed to vary between 0.28 and 0.35.

Parameters:

<MIndex> float
Range: 0.28 to 0.35
Increment: 0.01

Example: BB:BTO:PTYP DH1
sets the packet type.
BB:BTO:DTT:TABL LONG
enters the table type
BB:BTO:DTT:TABL:LONG:SET2:MIND 0.3
enters a modulation index.

Manual operation: See ["Modulation Index"](#) on page 48

[:SOURce<hw>]:BB:BTOoth:DTTTest:TABLE:LONG:SET<ch>:STATe <State>

The command activates or deactivates the corresponding parameter set for the long table. If a set deactivated, its parameters are skipped in the sequence. Instead, the next active set is used.

For Basic Rate packets, each set applies to 20ms of signal.

Parameters:

<State> 0 | 1 | OFF | ON

Example: BB:BTO:PTYP DH1
sets the packet type.
BB:BTO:DTT:TABL LONG
enters the table type
BB:BTO:DTT:TABL:LONG:SET2:STAT ON
activates the set 2 in the long table.

Manual operation: See ["State"](#) on page 47

[:SOURce<hw>]:BB:BTOoth:DTTTest:TABLE:LONG:SET<ch>:STERror <StError>

The command enters a symbol timing error in ppm.

The Symbol Timing Error modifies the symbol clock frequency by the set amount.

Parameters:

<StError> float
Range: -150 ppm to 150 ppm
Increment: 1 ppm

Example: BB:BTO:PTYP DH1
sets the packet type.
BB:BTO:DTT:TABL LONG
enters the table type
BB:BTO:DTT:TABL:LONG:SET2:STER -20
enters a symbol timing error.

Manual operation: See ["Symbol Timing Error"](#) on page 48

**[:SOURce<hw>]:BB:BTOoth:DTTTest:TABLE:SHORT:SET<ch>:CFOffset
<CfOffset>**

The command enters a carrier frequency offset.

The carrier frequency offset shows the deviation of the transmitted initial center frequency from carrier frequency.

Parameters:

<CfOffset> float
 Range: -150 kHz to 150 kHz
 Increment: 1 kHz

Example:

```
BB:BTO:PTYP DH1
sets the packet type.
BB:BTO:DTT:TABL SHOR
enters the table type
BB:BTO:DTT:TABL:SHOR:SET2:CFOF 65
enters a carrier frequency offset.
```

Manual operation: See ["Carrier Frequency Offset kHz"](#) on page 47

[[:SOURce<hw>]:BB:BTOoth:DTTtest:TABLE:SHORT:SET<ch>:STATe <State>

The command activates or deactivates the corresponding parameter set in the short table. If a set deactivated, its parameters are skipped in the sequence. Instead, the next active set is used.

For EDR packets, each set applies to 20 packets.

Parameters:

<State> 0 | 1 | OFF | ON
 *RST: ON

Example:

```
BB:BTO:PTYP DH1
sets the packet type.
BB:BTO:DTT:TABL SHOR
enters the table type
BB:BTO:DTT:TABL:SHOR:SET2:STAT ON
activates the set 2 in the short table.
```

Manual operation: See ["State"](#) on page 47

[[:SOURce<hw>]:BB:BTOoth:DTTtest:TABLE:SHORT:SET<ch>:STERror <StError>

The command enters a symbol timing error in ppm.

The Symbol Timing Error modifies the symbol clock frequency by the set amount.

Parameters:

<StError> float
 Range: -150 ppm to 150 ppm
 Increment: 1 ppm

Example: BB:BTO:PTYP DH1
sets the packet type.
BB:BTO:DTT:TABL SHOR
enters the table type
BB:BTO:DTT:TABL:SHOR:SET2:STER 20
enters a symbol timing error.

Manual operation: See ["Symbol Timing Error"](#) on page 48

4.9 LE Commands

[SOURce<hw>]:BB:BT00th:BCRole.....	105
[SOURce<hw>]:BB:BT00th:BMODE.....	106
[SOURce<hw>]:BB:BT00th:DTTest:TPConfiguration:TPInterval.....	106
[SOURce<hw>]:BB:BT00th:DTTest:TPConfiguration:UPLength.....	106
[SOURce<hw>]:BB:BT00th:DTTest:TPConfiguration:UPSource.....	106
[SOURce<hw>]:BB:BT00th:ECONfiguration:ACTable.....	107
[SOURce<hw>]:BB:BT00th:ECONfiguration:DCTable.....	107
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:DCMTable.....	107
[SOURce<hw>]:BB:BT00th:ECONfiguration:ACTable:SET<ch>:STATE.....	108
[SOURce<hw>]:BB:BT00th:ECONfiguration:DCTable:SET<ch>:STATE.....	108
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:DCMTable:SET<ch>:STATE.....	108
[SOURce<hw>]:BB:BT00th:ECONfiguration:AEDelay.....	108
[SOURce<hw>]:BB:BT00th:ECONfiguration:ADInterval.....	108
[SOURce<hw>]:BB:BT00th:ECONfiguration:AEInterval.....	109
[SOURce<hw>]:BB:BT00th:ECONfiguration:APIInterval.....	109
[SOURce<hw>]:BB:BT00th:ECONfiguration:LCMode.....	109
[SOURce<hw>]:BB:BT00th:ECONfiguration:LTKey.....	110
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:AADress.....	110
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:ACID.....	110
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:ACASigned.....	110
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:SCASigned.....	110
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:SCID.....	110
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:ICASigned.....	110
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:ICID.....	110
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:CID.....	111
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:CINstant.....	111
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:CINTerval.....	112
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:CIValue.....	112
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:DATA.....	112
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:DATA:DPATtern.....	113
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:DATA:DSElection.....	113
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:DLENGth.....	114
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:DWHitening.....	114
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:ECODE.....	114
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:EDIVersifier.....	115
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:FSLength.....	115
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:HLENGth.....	115

[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:LCTimeout.....	116
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:MIVector.....	116
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:SIVector.....	116
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:MSKD.....	116
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:SSKD.....	116
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:NCInterval.....	117
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:NLCTimeout.....	117
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:NSLatency.....	118
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:NSValue.....	118
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:NWOffset.....	118
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:NWSize.....	119
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:TAType.....	119
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:RAType.....	119
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:RVEctor.....	119
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:SCACcuracy.....	120
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:SLATency.....	120
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:SSValue.....	120
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:SVNumber.....	121
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:UTYPE.....	121
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:VNUMber.....	121
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:WOffset.....	122
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:WSIZe.....	122
[SOURce<hw>]:BB:BT00th:ECONfiguration:PNUMber.....	122
[SOURce<hw>]:BB:BT00th:ECONfiguration:SINTerval.....	123
[SOURce<hw>]:BB:BT00th:ECONfiguration:SWINdow.....	123
[SOURce<hw>]:BB:BT00th:ECONfiguration:WOINfo?.....	123
[SOURce<hw>]:BB:BT00th:ECONfiguration:WSINfo?.....	124
[SOURce<hw>]:BB:BT00th:UPTYpe.....	124
[SOURce<hw>]:BB:BT00th:USLength.....	125

[SOURce<hw>]:BB:BT00th:BCRole <BcRole>

Determines the controller role. Depending on the selected channel type different roles are assigned to the controller. For channel type "Data", master or slave can be assigned. If channel type "Advertising" is selected, the parameter is read only and displayed directly above the graph.

Parameters:

<BcRole> MASTER | SLAVE

MASTER
Assigns master role to the controller.

SLAVE
Selects slave as controller role.

*RST: MASTER

Example:

SOUR:BB:BT0:BCR MAST
master as controller role.
SOUR:BB:BT0:BCR SLAV
slave as controller role.

Manual operation: See ["Bluetooth Controller Role"](#) on page 28

[[:SOURce<hw>]:BB:BTOoth:BMODE <BMode>

Determines the digital Bluetooth standard. Basic Rate + EDR or Bluetooth Low Energy are available.

Parameters:

<BMode> BASic | BLENergy

BASic

Selects the Bluetooth mode Basic Rate + EDR.

BLENergy

Selects the Bluetooth mode Low Energy.

*RST: BASic

Example:

SOUR:BB:BTO:BMOD BLEN

Bluetooth mode Low Energy.

Manual operation: See ["Bluetooth Mode"](#) on page 18

[[:SOURce<hw>]:BB:BTOoth:DTTest:TPConfiguration:TPInterval <TpInterval>

Sets the time interval between two consecutive test packets, with regard to the starting points.

Parameters:

<TpInterval> float

Range: 0.625 to 12.5

Increment: 0.001

*RST: 0.625

Example:

SOUR:BB:BTO:DTT:TPC:TPIN 1.0

sets a time interval of 1.0.

Manual operation: See ["Packet Interval"](#) on page 44

[[:SOURce<hw>]:BB:BTOoth:DTTest:TPConfiguration:UPLength <UpLength>

Sets the payload length.

Parameters:

<UpLength> integer

Range: 1 to 37

*RST: 1

Example:

SOUR:BB:BTO:DTT:TPC:UPL 37

sets a payload length of 37.

Manual operation: See ["Payload Length"](#) on page 45

[[:SOURce<hw>]:BB:BTOoth:DTTest:TPConfiguration:UPSource <UpSource>

Selects the data source used for the payload test packets.

Parameters:

<UpSource> PN09 | PAT1 | PAT2 | PN15 | PAT3 | PAT4 | PAT5 | PAT6

PN9 / PN15

Select a PRBS-modulated data sequence (PRBS = pseudo random binary sequence) for testing.

PAT1 ... PAT6

Pattern is predefined.

*RST: PN09

Example:

SOUR:BB:BTO:DTT:TPC:UPS PN09
PRBS-modulated data sequence for testing.

Manual operation: See ["Payload Type"](#) on page 45

[SOURce<hw>]:BB:BTOoth:ECONfiguration:ACTable <AcTable>

[SOURce<hw>]:BB:BTOoth:ECONfiguration:DCTable <DcTable>

**[SOURce<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:DCMTTable
<DcmTable>**

The data channel map table indicates the entire data set of the channels in a table.

Every channel is represented with bit positioned as per the data channel index. LSB represents data channel index 0 and the bit in position 36 represents data channel index 36.

If the channel is used channel its bit is to be set to '1'. Bit value '0' indicates that the channel is unused.

The bits in positions 37, 38 and 39 shall be set to zero upon transmission and ignored upon receipt.

Note: This parameter is relevant for data event and advertising frame configuration with the packet types:

- CHANNEL_MAP_REQ
- CONNECT_REQ

Parameters:

<DcmTable> string

Example:

SOUR:BB:BTO:ECON:ACTable NOT
Entire data set for Advertising Channel Table
SOUR:BB:BTO:ECON:DCT NOT
Entire data set for Data Channel Table
SOUR:BB:BTO:ECON:PCON:DCMT NOT
Entire data set for Channel Map Table

Manual operation: See ["Channel Table"](#) on page 34

```
[ :SOURce<hw>]:BB:BTOoth:ECONfiguration:ACTable:SET<ch>:STATe <State>
[ :SOURce<hw>]:BB:BTOoth:ECONfiguration:DCTable:SET<ch>:STATe <State>
[ :SOURce<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:DCMTable:
    SET<ch>:STATe <State>
```

Indicates used and unused data channels.

Note: This parameter is relevant for data event and advertising frame configuration with the packet types:

- CHANNEL_MAP_REQ
- CONNECT_REQ

Parameters:

<State> 0 | 1 | OFF | ON
 *RST: ON

Example:

```
SOUR:BB:BTO:ECON:ACT:SET2:STAT ON
State in Advertising Channel Table
SOURce:BB:BTO:ECON:DCT:SET1:STAT ON
State in Data Channel Table
SOUR:BB:BTO:ECON:PCON:DCMT:SET1:STAT ON
State in Data Channel Map Table
```

Manual operation: See "[Channel Table](#)" on page 34

```
[ :SOURce<hw>]:BB:BTOoth:ECONfiguration:AEDelay <AeDelay>
```

Sets a time delay between the start times of two consecutive advertising events. The value is added to the advertising event interval.

Parameters:

<AeDelay> float
 Range: 0.0 to 10.0
 *RST: 0.0

Example:

```
SOUR:BB:BTO:ECON:AED 10.0
sets a time delay of 10.0 s.
```

Manual operation: See "[Advertising Event Delay](#)" on page 31

```
[ :SOURce<hw>]:BB:BTOoth:ECONfiguration:ADInterval <AdInterval>
```

Sets the time interval between two consecutive advertising events for packet type "ADV_DIRECT_IND".

Parameters:

<AdInterval> float
 Range: 0.9 to 3.0
 *RST: 3.0

Example:

```
SOUR:BB:BTO:ECON:ADIN 1.3
sets a time interval of 1.3.
```

Manual operation: See ["Advertising Event Interval"](#) on page 31

[[:SOURce<hw>]:BB:BTOoth:ECONfiguration:AEInterval <AeInterval>

Sets the time interval between two consecutive advertising events, with regard to the starting points.

Parameters:

<AeInterval> float
 Range: 20.0 to 10240.0
 *RST: 20.0

Example: SOUR:BB:BTO:ECON:AEIN 20.0
 sets a time interval of 20.0.

Manual operation: See ["Advertising Event Interval"](#) on page 31

[[:SOURce<hw>]:BB:BTOoth:ECONfiguration:APInterval <ApInterval>

Sets the time interval between packets starting points of two consecutive packets in the advertising channel.

Parameters:

<ApInterval> float
 Range: 1.3 to 1.5
 *RST: 1.3

Example: SOUR:BB:BTO:ECON:APIN 1.3
 sets a time interval of 1.3.

Manual operation: See ["Advertising Packet Interval"](#) on page 31

[[:SOURce<hw>]:BB:BTOoth:ECONfiguration:LCMode <LcMode>

Select the link layer connection mode. In order to provide safe transmission of payload data, the data in the packet can be encrypted. If activated, the payload data follows MIC (Message authentication Code).

Parameters:

<LcMode> UENC | ENC
UENC
 Payload data is transmitted without encoding.
ENC
 The link layer connection runs in encrypted mode.
 *RST: UENC

Example: SOUR:BB:BTO:ECON:LCM UENC
 without encoding.
 SOUR:BB:BTO:ECON:LCM ENC
 in encrypted mode.

Manual operation: See ["LL Connection Mode"](#) on page 33

[:SOURce<hw>]:BB:BT0oth:ECONfiguration:LTKey <LtKey>

Indicates the time the controller needs to receive the long term key from the host. After this time, the controller is ready to enter into the last phase of encryption mode setup.

Parameters:

<LtKey> integer

Example:

```
SOUR:BB:BT0:ECON:LCM ENC
SOUR:BB:BT0:ECON:LTK
#H00000000000000000000000000000000,128
In encrypted mode, the code can be edited.
```

Manual operation: See ["Long Term key \(hex\)"](#) on page 34

**[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:AAddress
<AAddress>**

Sets the access address of the link layer connection (32-bit string).

Parameters:

<AAddress> integer

Example:

```
SOUR:BB:BT0:ECON:PCON:AADD #H00000000,32
sets an access address.
```

Manual operation: See ["Access Address"](#) on page 36

[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:ACID <Acid>

**[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:ACASigned
<AcAssigned>**

**[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:SCASigned
<ScAssigned>**

[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:SCID <Scid>

**[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:ICASigned
<IcAssigned>**

[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:ICID <Icid>

Sets the advertiser's device address. In bluetooth low energy systems all the transmissions start with an 8 bit preamble followed by an access address. The access address is composed of a the parts "Company_Id" (LSB) and the "Company_assigned" (MSB). Beside the address fields the notation is given. For advertising channel packets the format of the device address differs, depending on the selected address type.

- "Public Address Types"

The public address is given from the registration authority IEEE and is composed of:

 - LSB: 24 bits = company_assigned
 - MSB: 24 bits = company_id
- "Private Address Type"

A private address is optional and composed of:

 - LSB: 24 bits = hash

- MSB: 24 bits = random

Parameters:

<Icid> integer

Example:

```
SOUR:BB:BTO:ECON:PCON:ACID #H000000,24
SOUR:BB:BTO:ECON:PCON:ACAS #H000000,24
Company_Assigned and Company_Id in Advertiser's Device
Address)
SOUR:BB:BTO:ECON:PCON:SCAS #H000000,24
SOUR:BB:BTO:ECON:PCON:SCID #H000000,24
Company_Assigned and Company_Id in Scanners Device
Address)
SOUR:BB:BTO:ECON:PCON:ICAS #H000000,24
SOUR:BB:BTO:ECON:PCON:ICID #H000000,24
Company_Assigned and Company_Id in Initiators Device
Address)
```

Manual operation: See ["Controller's Device Addr"](#) on page 41

[[:SOURce<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:CID <Cid>

Sets the company identifier of the manufacturer of the Bluetooth Controller. A 16 bit value is set.

Note: This parameter is relevant for data frame configuration and for the packet type LL_VERSION_IND.

Parameters:

<Cid> integer

*RST: 0

Example:

```
SOUR:BB:BTO:ECON:PCON:CID #H0000,16
sets the company ID.
```

Manual operation: See ["Company ID"](#) on page 43

**[[:SOURce<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:CINstant
<CInstant>**

Sets a connection instant for indicating the connection event at which the new connection parameters are taken in use.

Parameters:

<CInstant> float

Range: 0 to 1000

*RST: 0

Example:

```
SOUR:BB:BTO:ECON:PCON:CINS 2
sets a connection instant.
```

Manual operation: See ["Connection Instant"](#) on page 39

[[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:CINterval
<CInterval>

(for data event and advertising frame configuration with the packet type CONNECT_REQ)

Set the time interval between the start points of two consecutive connection events.

Parameters:

<CInterval> float
 Range: 7.5 to 4000.0
 *RST: 7.5

Example:

SOUR:BB:BT0:UPTY CREQ
 sets packet type CONNECT_REQ
 SOUR:BB:BT0:ECON:PCON:CINT 7.5
 sets a time interval.

Manual operation: See ["Connection Event Interval"](#) on page 32

[[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:CIValue <CIValue>

Sets the initialization value for the CRC (Cyclic Redundary Check, 24 bits) calculation. A packet has been received correctly, when it has passed the CRC check.

Parameters:

<CIValue> integer

Example:

SOUR:BB:BT0:ECON:PCON:CIV #H000000,24
 sets the initialization value for the CRC.

Manual operation: See ["CRC Initial"](#) on page 42

[[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:DATA <Data>

Selects the data source used for the payload.

Parameters:

<Data>

ALL0 | ALL1 | PATtern | PN09 | PN11 | PN15 | PN16 | PN20 |
PN21 | PN23 | DLISt**All 0 / All 1**

0 data and 1 data is generated internally.

Pattern

Pattern is user definable.

PN xxPseudo-random noise sequence. XX can be equal to 9, 11, 15,
16, 20, 21, 23.**Data List**Internal data from a programmable data list is used. The data list
can be generated by the Data List Editor or generated exter-
nally.

*RST: PN09

Example:

```
SOUR:BB:BTO:ECON:PCON:DATA ALL0 | ALL1
SOUR:BB:BTO:ECON:PCON:DATA PATT
SOUR:BB:BTO:ECON:PCON:DATA:DPAT #H3F,8
SOUR:BB:BTO:ECON:PCON:DATA PN09
SOUR:BB:BTO:ECON:PCON:DATA DLIS
SOUR:BB:BTO:ECON:PCON:DATA:DSEL 'bluetooth-le'
```

Manual operation: See ["Data Source"](#) on page 38

**[[:SOURce<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:DATA:DPATtern
<DPattern>**

The command selects the data for a pattern.

Parameters:

<DPattern>

integer

*RST: 0

Example:

```
BB:BTO:ECON:PCON:DATA PATT
sets the data type.
BB:BTO:ECON:PCON:DATA:DPAT #B010101, 6
selects the data for a pattern
```

Manual operation: See ["Data Source"](#) on page 38

**[[:SOURce<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:DATA:DSELection
<DSelection>**

The command selects data list file.

Parameters:

<DSelection>

string

Example: BB:BTO:ECON:PCON:DATA DLIS
selects the data type.
BB:BTO:ECON:PCON:DSEL Bluetooth-le
selects the file for the data.

Manual operation: See ["Data Source"](#) on page 38

[:SOURce<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:DLENgth
<DLength>

Enters the payload data length in bytes.

Parameters:

<DLength> float
Range: 0 to 31
*RST: 0

Example: SOUR:BB:BTO:ECON:PCON:DLEN 31
payload data length is 31 bytes.

Manual operation: See ["Data Length"](#) on page 38

[:SOURce<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:DWHitening
<DWhitening>

Activates or deactivates the Data Whitening. Evenly distributed white noise is ideal for the transmission and real data can be forced to look similar to white noise with different methods called Data Whitening. Applied to the PDU and CRC fields of all packet types, whitening is used to avoid long equal sequences in the data bit stream.

Parameters:

<DWhitening> 0 | 1 | OFF | ON
*RST: OFF

Example: BB:BTO:ECON:PCON:DWH ON
activates data whitening.

Manual operation: See ["Data Whitening"](#) on page 36

[:SOURce<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:ECODE <ECode>

Sets the error code value to inform the remote device why the connection is about to be terminated in case of LL_TERMINATE_IND packet. On the other hand, this parameter for LL_REJECT_IND packet is used for the reason a request was rejected. A 8 bit value is set.

Note: This parameter is relevant for data frame configuration and the packet type:

- LL_TERMINATE_IND
- LL_REJECT_IND

Parameters:

<ECode> integer
 *RST: 0

Example:

SOUR:BB:BTO:ECON:PCON:ECOD #H00,8
 sets the error code

Manual operation: See ["Error Code"](#) on page 43

[:SOURce<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:EDIVersifier
 <EDiversifier>

Sets the encrypted diversifier of the master for device identification. The parameter is an initialization vector provided by the Host in the HCI_ULP_Start_Encryption command.

Parameters:

<EDiversifier> integer

Example:

SOUR:BB:BTO:ECON:PCON:EDIV #H0000, 16
 sets the encrypted diversifier of the master.

Manual operation: See ["Encrypted DIVersifier \(hex\)"](#) on page 40

[:SOURce<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:FSLength
 <FsLength>

Enables that the feature set length is indicated. FeatureSet indicates whether the Controller features are used or not. All the data in FeatureSet is RFU(zero).

Parameters:

<FsLength> float
 Range: 1 to 26
 *RST: 26

Example:

SOUR:BB:BTO:ECON:PCON:FSL 12
 feature set length is 12.

Manual operation: See ["Feature Set Length"](#) on page 41

[:SOURce<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:HLENgth
 <HLength>

(for data event and advertising frame configuration with the packet type CONNECT_REQ)

Sets the difference from the current channel to the next channel. The master and slave devices determine the data channel in use for every connection event from the channel map. Hop_length is set for the LL connection and communicated in the CONNECT_REQ and CHANNEL_MAP_REQ packets.

Parameters:

<HLength> integer
 Range: 5 to 16
 *RST: 5

Example:

SOUR:BB:BTO:UPTY CREQ
 sets packet type CONNECT_REQ
 SOUR:BB:BTO:ECON:PCON:HLEN 10
 hop length is 10.

Manual operation: See "[Hop Length](#)" on page 40

**[[:SOURce<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:LCTimeout
 <LcTimeout>**

(for data event and advertising frame configuration with the packet type:

CONNECT_REQ

Defines the maximum time between two correctly received Bluetooth Low Energy packets in the LL connection before the connection is considered lost.

Parameters:

<LcTimeout> float
 Range: 100.0 to 32000.0
 *RST: 100.0

Example:

SOUR:BB:BTO:UPTY CREQ
 sets packet type CONNECT_REQ
 SOUR:BB:BTO:ECON:PCON:LCT 150
 LL Connection Timeout is 150.

Manual operation: See "[LL Connection Timeout](#)" on page 39

**[[:SOURce<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:MIVector
 <MiVector>**

**[[:SOURce<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:SIVector
 <SiVector>**

Sets the master's or the slave's portion of the initialization vector (IVm/IVs).

Parameters:

<SiVector> integer

Example:

SOUR:BB:BTO:ECON:PCON:MIV #H0000000000000000, 32
 (Master).
 SOUR:BB:BTO:ECON:PCON:SIV #H0000000000000000, 32
 (Slave).

**[[:SOURce<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:MSKD <Mskd>
 [[:SOURce<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:SSKD <Sskd>**

Sets the master's or the slave's portion of the session key diversifier (SKDm/SKDd).

Parameters:

<Sskd> integer

Example:

SOUR:BB:BTO:ECON:PCON:MSKD
 #H0000000000000000,64
 (Master).
 SOUR:BB:BTO:ECON:PCON:SSKD
 #H0000000000000000,64
 (Slave).

[:SOURce<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:NCInterval
 <NcInterval>

(for data event and advertising frame configuration with the packet type CONNECTION_UPDATE_REQ)

Set the time interval between the start points of two consecutive connection events.

Parameters:

<NcInterval> float
 Range: 7.5 to 4000.0
 *RST: 7.5

Example:

SOUR:BB:BTO:UPTY CUR
 sets packet type CONNECTION_UPDATE_REQ
 SOUR:BB:BTO:ECON:PCON:NCIN 7.5
 sets a time interval.

Manual operation: See ["Connection Event Interval"](#) on page 32

[:SOURce<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:NLCTimeout
 <NlcTimeout>

(for data event and advertising frame configuration with the packet type CONNECTION_UPDATE_REQ)

Defines the maximum time between two correctly received Bluetooth Low Energy packets in the LL connection before the connection is considered lost.

Parameters:

<NlcTimeout> float
 Range: 100.0 to 32000.0
 *RST: 100.0

Example:

SOUR:BB:BTO:UPTY CUR
 sets packet type CONNECTION_UPDATE_REQ
 SOUR:BB:BTO:ECON:PCON:NLCT 150
 LL Connection Timeout is 150.

Manual operation: See ["LL Connection Timeout"](#) on page 39

**[[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:NSLatency
<NSLatency>**

(for data event and advertising frame configuration with the packet type CONNECTION_UPDATE_REQ)

Sets a number of consecutive connection events the slave can ignore for asymmetric link layer connections.

Parameters:

<NSLatency> integer
 Range: 0 to 1000
 *RST: 0

Example:

SOUR:BB:BT0:UPTY CUR
 sets packet type CONNECTION_UPDATE_REQ
 SOUR:BB:BT0:ECON:PCON:NSL 10
 sets the number of consecutive connection events.

Manual operation: See ["Slave Latency"](#) on page 39

**[[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:NSValue
<NSValue>**

Sets the start value of the next expected packet from the same device in the LL connection ("N"ext"E"xpected "S"equence"N"umber). This parameter can be set in the first event. From the second event this field is not indicated.

Parameters:

<NSValue> integer
 Range: 0 to 1
 *RST: 1

Example:

SOUR:BB:BT0:ECON:PCON:NSV 1
 Start Value is 1.

Manual operation: See ["NESN Start Value"](#) on page 37

**[[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:NWOffset
<NWOffset>**

(for data event and advertising frame configuration with the packet type CONNECTION_UPDATE_REQ)

Sets the start point of the transmit window.

Parameters:

<NWOffset> float
 Range: 0.0 to 5000.0
 *RST: 0.0

Example: SOUR:BB:BTO:UPTY CUR
 sets packet type CONNECTION_UPDATE_REQ
 SOUR:BB:BTO:ECON:PCON:NWOF 800.0
 sets the start point of the transmit window.

Manual operation: See ["Transmit Window Offset"](#) on page 42

[SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:NWSize <NwSize>

(for data event and advertising frame configuration with the packet type CONNECTION_UPDATE_REQ)

Sets the size of the transmit window, regarding to the start point.

Parameters:

<NwSize> float
 Range: 1.25 to 10.0
 *RST: 1.25

Example: SOUR:BB:BTO:UPTY CUR
 sets packet type CONNECTION_UPDATE_REQ
 SOUR:BB:BTO:ECON:PCON:NWS 8.0
 sets the size of the transmit window.

Manual operation: See ["Transmit Window Size"](#) on page 42

[SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:TAType <TaType>
[SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:RAType <RaType>

Selects the address type of the controller device. Depending on the Bluetooth controller role either Tx or Rx or both address types are assigned. Subdivided into private and random, a Bluetooth LE device address consists of 48 bits. The format of the device address differs depending on the selected address type.

Parameters:

<RaType> PUBLIC | RANDom
PUBLIC
 Allocates a unique 48 bit address to each bluetooth LE device. The public address is given from the registration authority IEEE.
RANDom
 Allocates a 48 bit address to each bluetooth LE device. A random address is optional.
 *RST: PUBLIC

Example: SOUR:BB:BTO:ECON:PCON:TATY PUBL
 SOUR:BB:BTO:ECON:PCON:RATY RAND

[SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:RVEctor <RVector>

Sets the random vector of the master for device identification. The parameter is an initialization vector provided by the Host in the HCI_ULP_Start_Encryption command.

Parameters:**<RVector>** integer**Example:**

SOUR:BB:BTO:ECON:PCON:RVEC
 #H0000000000000000,64

sets the random vector of the master.

Manual operation: See ["Random Vector \(hex\)"](#) on page 40

**[:SOURce<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:SCACcuracy
 <ScAccuracy>**

Defines the master's clock accuracy with specified encoding. This parameter is used by the slave to determine required listening windows in the LL connection. It is a controller design parameter known by the Controller.

Parameters:

<ScAccuracy> SCA0 | SCA1 | SCA2 | SCA3 | SCA4 | SCA5 | SCA6 | SCA7
 *RST: SCA0

Example:

SOUR:BB:BTO:ECON:PCON:SCAC SCA1
 sets the encoding value

Manual operation: See ["Sleep Clock Accuracy"](#) on page 43

**[:SOURce<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:SLATency
 <SLatency>**

(for data event and advertising frame configuration with the packet type CONNECT_REQ)

Sets a number of consecutive connection events the slave can ignore for asymmetric link layer connections.

Parameters:

<SLatency> integer
 Range: 0 to 1000
 *RST: 0

Example:

SOUR:BB:BTO:UPTY CREQ
 sets packet type CONNECT_REQ
 SOUR:BB:BTO:ECON:PCON:SLAT 10
 sets the number of consecutive connection events.

Manual operation: See ["Slave Latency"](#) on page 39

[:SOURce<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:SSValue <SsValue>

Sets the sequence number of the packet. This parameter can be set in the first event. From the second event this field is not indicated.

Parameters:

<SsValue> integer
 Range: 0 to 1
 *RST: 0

Example:

SOUR:BB:BTO:ECON:PCON:SSV 1
 sets the sequence number of the packet.

Manual operation: See ["SN Start Value"](#) on page 37

[:SOURce<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:SVNumber
 <SvNumber>

Sets a unique value for each implementation or revision of an implementation of the Bluetooth Controller. A 16 bit value is set.

Note: This parameter is relevant for data frame configuration and for the packet type: LL_VERSION_IND.

Parameters:

<SvNumber> integer
 *RST: 0

Example:

SOUR:BB:BTO:ECON:PCON:SVN #H0000,16
 sets the sub version number

Manual operation: See ["Sub Version Number"](#) on page 44

[:SOURce<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:UTYPE <UType>

Enables that an invalid control packet is indicated. The CtrType field indicates the value of the LL control packet that caused the transmission of this packet.

Parameters:

<UType> integer

Example:

SOUR:BB:BTO:ECON:PCON:UTYP #H8, 00
 enables that an invalid control packet is indicated.

Manual operation: See ["Unknown Type \(hex\)"](#) on page 41

[:SOURce<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:VNUMBER
 <VNumber>

Sets the company identifier of the manufacturer of the Bluetooth Controller. A 8 bit value is set.

Note: This parameter is relevant for data frame configuration and for the packet type LL_VERSION_IND.

Parameters:

<VNumber> integer
 *RST: 0

Example: `SOUR:BB:BTO:ECON:PCON:VNUM #H00,8`
sets the version number

Manual operation: See ["Version Number"](#) on page 43

[:SOURce<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:WOFFset
<WOffset>

(for data event and advertising frame configuration with the packet type CONNECT_REQ)

Sets the start point of the transmit window.

Parameters:

<WOffset> float
Range: 0.0 to 5000.0
*RST: 0

Example: `SOUR:BB:BTO:UPTY CREQ`
sets packet type CONNECT_REQ
`SOUR:BB:BTO:ECON:PCON:WOFF 800.0`
sets the start point of the transmit window.

Manual operation: See ["Transmit Window Offset"](#) on page 42

[:SOURce<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:WSIZe <WSize>

(for data event and advertising frame configuration with the packet type CONNECT_REQ)

Sets the size of the transmit window, regarding to the start point.

Parameters:

<WSize> float
Range: 1.25 to 10.0
*RST: 1.25

Example: `SOUR:BB:BTO:UPTY CREQ`
sets packet type CONNECT_REQ
`SOUR:BB:BTO:ECON:PCON:WSIZ 8.0`
sets the size of the transmit window.

Manual operation: See ["Transmit Window Size"](#) on page 42

[:SOURce<hw>]:BB:BTOoth:ECONfiguration:PNUMber <PNumber>

Sets the number of Tx packets per event. Each connection contains at least one data channel packet. The maximum number of packets per event is determined by the duration of the connection event interval.

Parameters:

<PNumber> float
 Range: 1 to 2580
 *RST: 1

Example:

SOUR:BB:BTO:ECON:PNUM 2580
 sets the number of Tx packets per event.

Manual operation: See ["No. of Tx Packets per Event"](#) on page 32

[SOURce<hw>]:BB:BTOoth:ECONfiguration:SINTerval <SInterval>

Sets the time interval between the starting points of two consecutive windows during which the scanner is operating in an advertising channel.

Parameters:

<SInterval> float
 Range: 2.5 to 10240.0
 *RST: 2.5

Example:

SOUR:BB:BTO:ECON:SINT 3.5
 sets the time interval.

Manual operation: See ["Scan Interval"](#) on page 31

[SOURce<hw>]:BB:BTOoth:ECONfiguration:SWINdow <SWindow>

Sets the length of the window during which the scanner is operating in the advertising channel. Note that the scan window is less or equal to the value of the scan interval.

Parameters:

<SWindow> float
 Range: 2.5 to 10240.0
 *RST: 2.5

Example:

SOUR:BB:BTO:ECON:SWIN 2.5
 sets the length of the window.

Manual operation: See ["Scan Window"](#) on page 31

[SOURce<hw>]:BB:BTOoth:ECONfiguration:WOINfo?

(for data event and advertising frame configuration with the packet type CONNECT_REQ)

Queries the start point of the transmit window.

Return values:

<WoInfo> string

Example: SOUR:BB:BTO:UPTY CREQ
 sets packet type CONNECT_REQ
 SOUR:BB:BTO:ECON:PCON:WOIN?
 queries the start point of the transmit window.

Usage: Query only

Manual operation: See ["Transmit Window Offset"](#) on page 32

[SOURce<hw>]:BB:BTOoth:ECONfiguration:WSINfo?

(for data event and advertising frame configuration with the packet type CONNECT_REQ)

Queries the size of the transmit window, regarding to the start point.

Return values:

<WsInfo> string

Example: SOUR:BB:BTO:UPTY CREQ
 sets packet type CONNECT_REQ
 SOUR:BB:BTO:ECON:PCON:WSIN?
 queries the size of the transmit window.

Usage: Query only

Manual operation: See ["Transmit Window Size"](#) on page 32

[SOURce<hw>]:BB:BTOoth:UPTYPE <UpType>

Selects the packet type. The available packets depend on the selected channel type.

Parameters:

<UpType> AIND | ADIND | ANIND | SREQ | SRSP | CREQ | ADCind |
 DATA | CUREq | CMReq | TIND | EREQ | ERSP | SEReq |
 SERSp | URSP | FREQ | FRSP | TPACKet | PEReq | PERSp |
 VIND | RIND
 *RST: AIND

Example:

```

SOUR:BB:BTO:UPTY AIND
SOUR:BB:BTO:UPTY ADIN
SOUR:BB:BTO:UPTY ANIN
SOUR:BB:BTO:UPTY ADC
SOUR:BB:BTO:UPTY SREQ
SOUR:BB:BTO:UPTY SRSP
SOUR:BB:BTO:UPTY CREQ
SOUR:BB:BTO:UPTY TPAC

```

Channel type "Advertising"

```

SOUR:BB:BTO:UPTY DATA
SOUR:BB:BTO:UPTY CUR
SOUR:BB:BTO:UPTY CMR
SOUR:BB:BTO:UPTY TIND
SOUR:BB:BTO:UPTY LER
SOUR:BB:BTO:UPTY SERS
SOUR:BB:BTO:UPTY FREQ
SOUR:BB:BTO:UPTY TPAC

```

Channel type "Data" and Controller role "Master".

```

SOUR:BB:BTO:UPTY DATA
SOUR:BB:BTO:UPTY TIND
SOUR:BB:BTO:UPTY LERS
SOUR:BB:BTO:UPTY SER
SOUR:BB:BTO:UPTY SERS
SOUR:BB:BTO:UPTY URSP
SOUR:BB:BTO:UPTY FRSP
SOUR:BB:BTO:UPTY TPAC

```

Channel type "Data" and Controller role "Slave".

Manual operation: See ["Packet Type"](#) on page 27

[[:SOURce<hw>]:BB:BTOoth:USLength <UsLength>

Selects the number of frames or events depending on the packet type. The signal repeats after the specified number of frames/events.

For SCAN_REQ and CONNECT_REQ packet, the sequence length is expressed in "Frames".

For TERMINATE_IND packets, a default value according to the specification is given:

Master: 'SlaveLatency + 6'

Slave: '6'

For all other packet types the sequence length is expressed in "Events".

Parameters:

<UsLength>	float
Range:	1 to 1000
*RST:	1

Example:

```

SOUR:BB:BTO:USL 1000
selects the number of frames or events.

```

Manual operation: See ["Sequence Length"](#) on page 28

List of Commands

[SOURce<hw>]:BB:BT00th:BCRole.....	105
[SOURce<hw>]:BB:BT00th:BCText?.....	64
[SOURce<hw>]:BB:BT00th:BMODE.....	106
[SOURce<hw>]:BB:BT00th:CLIPping:LEVel.....	70
[SOURce<hw>]:BB:BT00th:CLIPping:MODE.....	70
[SOURce<hw>]:BB:BT00th:CLIPping:STATe.....	71
[SOURce<hw>]:BB:BT00th:CLOCK:MODE.....	86
[SOURce<hw>]:BB:BT00th:CLOCK:MULTiplier.....	87
[SOURce<hw>]:BB:BT00th:CLOCK:SOURce.....	87
[SOURce<hw>]:BB:BT00th:CLOCK:SYNChronization:EXECute.....	88
[SOURce<hw>]:BB:BT00th:CLOCK:SYNChronization:MODE.....	88
[SOURce<hw>]:BB:BT00th:CTYPe.....	65
[SOURce<hw>]:BB:BT00th:DTTest:DTTState.....	99
[SOURce<hw>]:BB:BT00th:DTTest:FDDeviation.....	99
[SOURce<hw>]:BB:BT00th:DTTest:FDRate.....	100
[SOURce<hw>]:BB:BT00th:DTTest:SPHase.....	100
[SOURce<hw>]:BB:BT00th:DTTest:STDefault.....	100
[SOURce<hw>]:BB:BT00th:DTTest:TABLE.....	100
[SOURce<hw>]:BB:BT00th:DTTest:TABLE:LONG:SET<ch>:CFOFset.....	101
[SOURce<hw>]:BB:BT00th:DTTest:TABLE:LONG:SET<ch>:MINDex.....	101
[SOURce<hw>]:BB:BT00th:DTTest:TABLE:LONG:SET<ch>:STATe.....	102
[SOURce<hw>]:BB:BT00th:DTTest:TABLE:LONG:SET<ch>:STERror.....	102
[SOURce<hw>]:BB:BT00th:DTTest:TABLE:SHORT:SET<ch>:CFOFset.....	102
[SOURce<hw>]:BB:BT00th:DTTest:TABLE:SHORT:SET<ch>:STATe.....	103
[SOURce<hw>]:BB:BT00th:DTTest:TABLE:SHORT:SET<ch>:STERror.....	103
[SOURce<hw>]:BB:BT00th:DTTest:TPConfiguration:TPINterval.....	106
[SOURce<hw>]:BB:BT00th:DTTest:TPConfiguration:UPLength.....	106
[SOURce<hw>]:BB:BT00th:DTTest:TPConfiguration:UPSSource.....	106
[SOURce<hw>]:BB:BT00th:ECONfiguration:ACTable.....	107
[SOURce<hw>]:BB:BT00th:ECONfiguration:ACTable:SET<ch>:STATe.....	108
[SOURce<hw>]:BB:BT00th:ECONfiguration:ADINterval.....	108
[SOURce<hw>]:BB:BT00th:ECONfiguration:AEDelay.....	108
[SOURce<hw>]:BB:BT00th:ECONfiguration:AEINterval.....	109
[SOURce<hw>]:BB:BT00th:ECONfiguration:APINterval.....	109
[SOURce<hw>]:BB:BT00th:ECONfiguration:DCTable.....	107
[SOURce<hw>]:BB:BT00th:ECONfiguration:DCTable:SET<ch>:STATe.....	108
[SOURce<hw>]:BB:BT00th:ECONfiguration:LCMode.....	109
[SOURce<hw>]:BB:BT00th:ECONfiguration:LTKey.....	110
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:AADdress.....	110
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:ACASsigned.....	110
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:ACID.....	110
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:CID.....	111
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:CINstant.....	111
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:CINterval.....	112
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:CIValue.....	112
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:DATA.....	112
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:DATA:DPATtern.....	113

[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:DATA:DSElection.....	113
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:DCMTable.....	107
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:DCMTable:SET<ch>:STATe.....	108
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:DLENgth.....	114
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:DWHitening.....	114
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:ECODE.....	114
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:EDIVersifier.....	115
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:FSLength.....	115
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:HLENgth.....	115
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:ICASSigned.....	110
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:ICID.....	110
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:LCTimeOut.....	116
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:MIVector.....	116
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:MSKD.....	116
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:NCINterval.....	117
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:NLCTimeOut.....	117
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:NSLatency.....	118
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:NSValue.....	118
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:NWOFFset.....	118
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:NWSize.....	119
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:RAType.....	119
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:RVEctor.....	119
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:SCACcuracy.....	120
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:SCASSigned.....	110
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:SCID.....	110
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:SIVector.....	116
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:SLATency.....	120
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:SSKD.....	116
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:SSValue.....	120
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:SVNumber.....	121
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:TAType.....	119
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:UTYPE.....	121
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:VNUMBER.....	121
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:WOFFset.....	122
[SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:WSIZE.....	122
[SOURce<hw>]:BB:BT00th:ECONfiguration:PNUMBER.....	122
[SOURce<hw>]:BB:BT00th:ECONfiguration:SINTerval.....	123
[SOURce<hw>]:BB:BT00th:ECONfiguration:SWINdow.....	123
[SOURce<hw>]:BB:BT00th:ECONfiguration:WOINfo?.....	123
[SOURce<hw>]:BB:BT00th:ECONfiguration:WSINfo?.....	124
[SOURce<hw>]:BB:BT00th:FILTer:ILENgth.....	71
[SOURce<hw>]:BB:BT00th:FILTer:ILENgth:AUTO[STATe].....	72
[SOURce<hw>]:BB:BT00th:FILTer:MINdex.....	73
[SOURce<hw>]:BB:BT00th:FILTer:MTYPE.....	73
[SOURce<hw>]:BB:BT00th:FILTer:OSAMpling.....	72
[SOURce<hw>]:BB:BT00th:FILTer:OSAMpling:AUTO[STATe].....	72
[SOURce<hw>]:BB:BT00th:FILTer:PARAmeter:APCO25.....	73
[SOURce<hw>]:BB:BT00th:FILTer:PARAmeter:COsine.....	74
[SOURce<hw>]:BB:BT00th:FILTer:PARAmeter:FGAuss.....	74
[SOURce<hw>]:BB:BT00th:FILTer:PARAmeter:GAUSS.....	74

[SOURce<hw>]:BB:BT00th:FILTer:PARAmeter:LPAss.....	74
[SOURce<hw>]:BB:BT00th:FILTer:PARAmeter:PGAuss.....	75
[SOURce<hw>]:BB:BT00th:FILTer:PARAmeter:RCOSine.....	75
[SOURce<hw>]:BB:BT00th:FILTer:PARAmeter:SPHase.....	75
[SOURce<hw>]:BB:BT00th:FILTer:TYPE.....	71
[SOURce<hw>]:BB:BT00th:MSETtings:FDEVIation.....	73
[SOURce<hw>]:BB:BT00th:PCONfiguration:ACKNowledgeMent.....	91
[SOURce<hw>]:BB:BT00th:PCONfiguration:BDALap.....	91
[SOURce<hw>]:BB:BT00th:PCONfiguration:BDANap.....	91
[SOURce<hw>]:BB:BT00th:PCONfiguration:BDAUap.....	92
[SOURce<hw>]:BB:BT00th:PCONfiguration:CODEvice.....	92
[SOURce<hw>]:BB:BT00th:PCONfiguration:DATA.....	92
[SOURce<hw>]:BB:BT00th:PCONfiguration:DATA:DPATtern.....	93
[SOURce<hw>]:BB:BT00th:PCONfiguration:DATA:DSELEction.....	93
[SOURce<hw>]:BB:BT00th:PCONfiguration:DATA:VDPAtern.....	93
[SOURce<hw>]:BB:BT00th:PCONfiguration:DATA:VDSELEction.....	94
[SOURce<hw>]:BB:BT00th:PCONfiguration:DLENgth.....	94
[SOURce<hw>]:BB:BT00th:PCONfiguration:DSFPacket.....	94
[SOURce<hw>]:BB:BT00th:PCONfiguration:DWHitening.....	95
[SOURce<hw>]:BB:BT00th:PCONfiguration:EIRPacketfollows.....	95
[SOURce<hw>]:BB:BT00th:PCONfiguration:HFControl.....	96
[SOURce<hw>]:BB:BT00th:PCONfiguration:LTADdress.....	96
[SOURce<hw>]:BB:BT00th:PCONfiguration:PFControl.....	96
[SOURce<hw>]:BB:BT00th:PCONfiguration:PLENgth.....	97
[SOURce<hw>]:BB:BT00th:PCONfiguration:SNSValue.....	97
[SOURce<hw>]:BB:BT00th:PCONfiguration:SRMode.....	98
[SOURce<hw>]:BB:BT00th:PCONfiguration:VDATA.....	98
[SOURce<hw>]:BB:BT00th:PRAMping:FOFFset.....	89
[SOURce<hw>]:BB:BT00th:PRAMping:RFUNction.....	89
[SOURce<hw>]:BB:BT00th:PRAMping:ROFFset.....	89
[SOURce<hw>]:BB:BT00th:PRAMping:RTIME.....	90
[SOURce<hw>]:BB:BT00th:PRESet.....	65
[SOURce<hw>]:BB:BT00th:PTYPE.....	66
[SOURce<hw>]:BB:BT00th:SETTing:CATalog.....	66
[SOURce<hw>]:BB:BT00th:SETTing:DELeTe.....	66
[SOURce<hw>]:BB:BT00th:SETTing:LOAD.....	67
[SOURce<hw>]:BB:BT00th:SETTing:STORE.....	67
[SOURce<hw>]:BB:BT00th:SETTing:STORE:FAST.....	67
[SOURce<hw>]:BB:BT00th:SLENgth.....	68
[SOURce<hw>]:BB:BT00th:SRATe:VARiation.....	76
[SOURce<hw>]:BB:BT00th:STATe.....	68
[SOURce<hw>]:BB:BT00th:TMODE.....	68
[SOURce<hw>]:BB:BT00th:TRIGger:ARM:EXECute.....	76
[SOURce<hw>]:BB:BT00th:TRIGger:EXECute.....	77
[SOURce<hw>]:BB:BT00th:TRIGger:OBASeband:DELaY.....	78
[SOURce<hw>]:BB:BT00th:TRIGger:OBASeband:INHibit.....	78
[SOURce<hw>]:BB:BT00th:TRIGger:OUTPut:DELaY:FIXed.....	82
[SOURce<hw>]:BB:BT00th:TRIGger:OUTPut<ch>:DELaY.....	83
[SOURce<hw>]:BB:BT00th:TRIGger:OUTPut<ch>:DELaY:MAXimum?.....	83
[SOURce<hw>]:BB:BT00th:TRIGger:OUTPut<ch>:DELaY:MINimum?.....	83

[:SOURce<hw>]:BB:BT00th:TRIGger:OUTPut<ch>:MODE.....	84
[:SOURce<hw>]:BB:BT00th:TRIGger:OUTPut<ch>:OFFTime.....	85
[:SOURce<hw>]:BB:BT00th:TRIGger:OUTPut<ch>:ONTime.....	85
[:SOURce<hw>]:BB:BT00th:TRIGger:OUTPut<ch>:PATtern.....	85
[:SOURce<hw>]:BB:BT00th:TRIGger:OUTPut<ch>:PULSe:DIVider.....	85
[:SOURce<hw>]:BB:BT00th:TRIGger:OUTPut<ch>:PULSe:FREQuency?.....	86
[:SOURce<hw>]:BB:BT00th:TRIGger:RM0de.....	78
[:SOURce<hw>]:BB:BT00th:TRIGger:SLENgth.....	79
[:SOURce<hw>]:BB:BT00th:TRIGger:SLUNit.....	79
[:SOURce<hw>]:BB:BT00th:TRIGger:SOURce.....	80
[:SOURce<hw>]:BB:BT00th:TRIGger[:EXtErnal]:SYNChronize:OUTPut.....	77
[:SOURce<hw>]:BB:BT00th:TRIGger[:EXtErnal<ch>]:DELay.....	80
[:SOURce<hw>]:BB:BT00th:TRIGger[:EXtErnal<ch>]:INHibit.....	81
[:SOURce<hw>]:BB:BT00th:UPTYPE.....	124
[:SOURce<hw>]:BB:BT00th:USLength.....	125
[:SOURce<hw>]:BB:BT00th:VERSion.....	69
[:SOURce<hw>]:BB:BT00th:WAVEform:CREate.....	69
[:SOURce<hw>]:BB:BT00th[:TRIGger]:SEQuence.....	81

Index

A

Access Address	36
ACK	23
Acknowledgment	23
ACL+EDR	20
Activate Bluetooth Standard	16
Activate Dirty Transmitter Test	46
Advertiser	28
Advertiser's address type	37
Advertiser's device address	41
Advertising Channel Table	34
Advertising channel type	26
Advertising Event Delay	31
Advertising Event Interval	31
Advertising Packet Interval	31
Advertising Settings	
Advertising Event Delay	31
Advertising Event Interval	31
Advertising Packet Interval	31
Scan Interval	31
Scan Window	31
Test Packet Interval	44
Test Packet Payload Length	45
Transmit window offset	32, 42
Transmit window size	32, 42
All Data	21
Arm Trigger	19, 55, 76
Armed_Auto	81
Armed_Retrigger	81
Auto	81

B

B x T	73
Baseband Clipping	51
Baseband filter	49, 71
Baseband Filter	19
Basic Rate+EDR	18
Bluetooth LE Version	18
Bluetooth Mode	18
Bluetooth Mode - Basic Rate+EDR	18
Bluetooth Mode - Low Energy	18
Bluetooth Version	18
BxT	49

C

Carrier Frequency Offset (DTT)	47
CDMA Version	69
Center Frequency	34
Channel Index	34
Channel State	34, 108
Channel Type	26, 28, 34
Channel Type Advertising	26
Channel Type Data	26
Chip Rate Variation	76
Class of Device	25
Clipping	19
Clipping Level	51
Clipping Mode	52, 70
Clipping State	51
Clock Mode	61

Clock Multiplier	62
Clock Source	61
Company ID	43
Connection Event Interval	32, 38
Connection Settings	
Connection Event Interval	32, 38
LL Connection Mode	33
Long term key (hex)	34
No. of Tx Packets per Event	32
Selected data channel	34
Controller role master	28
Controller role slave	28
Controller role advertiser	28
Controller role initiator	28
Controller role scanner	28
Controller state	28
Controller's device address	41
Conventions	
SCPI commands	63
CRC Initial	42
Crest factor	51
Crest factor - Clipping	70
Current Range without Recalculation	60
Current Range Without Recalculation Maximum	83
Current Range Without Recalculation Minimum	83
Cut Off Frequency Factor	50

D

Data Channel Map Table	107
Data Channel Table	34
Data channel type	26
Data length	
Packet configuration	24
Data Length	
Packet configuration	38
Data List Management	17
Data source	
Packet configuration	23, 38
Voice Field	25
Data Whitening	22, 36, 114
Deactivate Bluetooth Standard	16
Deactivate Dirty Transmitter Test	46
Default settings	65
Default Settings	16
Delay - Marker	60, 83
Delay - Trigger	80
Device Address	22
Device's address type	37
Dirty Transmitter Test	
Carrier Frequency Offset	47
Frequency Drift Deviation	47
Frequency Drift Rate	47
Modulation Index	48
Start Phase	47
State (Long Table)	47
State (Short Table)	47
Symbol Timing Error	48
Dirty Transmitter Test State	46
Documentation Overview	5

E

Edit Data List	17
EIR Packet Follows	24
Encrypted DIVERSifier(hex)	40
Encrypted mode	33
Error Code	43
eSCO+EDR	20
Event Configuration	29
Channel Table	34
Execute Trigger	19, 56, 77
External Trigger Delay	80
External Trigger Inhibit	81

F

Fall Offset	53
Feature Set Length	41
File Manager	16
Filter	19, 49
Filter Parameter	73
Roll off	49
Filter Type	71
Filtering, Clipping	19
Fix marker delay to current range	60
Fix marker to dynamic range	82
Flow Control (Header)	22
Flow Control (Payload)	24
Frame Active Part Marker	59
Frame Configuration	29
Channel Table	34
Frame Start Marker	59
Frames – Sequence Length	28
Frequency Deviation	51
Frequency Drift Deviation	47
Frequency Drift Rate	47

G

Gauss FSK filter	49
Generate Waveform File	18
Global Trigger/Clock Settings	62

H

Hide Data Channel Mapping Table	40
Hide Data Channel Table	40
Hop Length	40

I

Impulse Length	50, 71
Impulse Length Auto State	72
Initialization Vector(hex)	41
Initiator	28
Initiator's address type	37
Initiator's device address	41

L

LAP	22
LL Connection Event Count	39
LL Connection Mode	33
LL Connection Timeout	39
Load Bluetooth Settings	16
Logical Transport Address	22
Long Term key (hex)	34

Low Energy	18
------------------	----

M

Manual Trigger	77
Marker	19
Marker Delay	60, 83
Marker Mode	59, 84
Master	28
Measured external clock	62
MIC (Message authentication Code)	33
Modulation Index	51
Modulation Index (DTT)	48
Modulation Type	50
Multiple Sample Clock Mode	61

N

NAK	23
NAP	22
NESN Start Value	37
No. of Tx Packets per Event	32
Nyquist filter	71

O

ON/OFF Ratio Marker	59, 85
Open File Manager	16
Oversampling	50
Oversampling Auto State	72
Oversamplingt	72

P

Packet Configuration	20, 32
Access Address	36
Acknowledgment	23
advertiser's address type	37
Center Frequency	34
Channel Index	34
Channel State	34
Channel Type	34
Class of Device	25
Company ID	43
Connection Event Interval	32, 38
CRC Initial	42
Data Whitening	22, 36
Device Address	22
EIR Packet Follows	24
Encrypted DIVERSifier(hex)	40
Error Code	43
Feature Set Length	41
Flow Control (Header)	22
Flow Control (Payload)	24
Hide Data Channel Mapping Table	40
Hide Data Channel Table	40
Hop Length	40
Initialization Vector(hex)	41
initiator's address type	37
Initiator's device address	41
LAP	22
LL Connection Event Count	39
LL Connection Timeout	39
Logical Transport Address	22
NAP	22
NESN Start Value	37
Packet Length	24

Random Vector(hex)	40
Scan Repetition Mode	24
Scanner's device address	41
SEQN Start Value	23
Show Data Channel Mapping Table	40
Show Data Channel Table	40
Simulation Mode	21
Slave Latency	39
Sleep Clock Accuracy	43
SN Start Value	37
Sub Version Number	44
UAP	22
Unknown Type(hex)	41
Version Number	43
Packet Configuration: Session Key iD(hex)	40
Packet Configuration:Channel State	108
Packet Configuration:Channel Table	107
Packet Configuration:Data Channel Map Table	107
Packet Configuration:Data Whitening	114
Packet Editor	21
Packet Length	24
Packet Structure	44
Packet Type	20, 27
Pattern Marker	59
Payload Type Test Packet	45
Power Ramping	19, 52
Private address type	37, 41
Public address type	37, 41
Pulse Marker	59

R

Ramp Function	52, 89
Ramp Time	53, 90
Random Vector(hex)	40
Recall BluetoothSettings	16
Recall Configuration	16
Restart Marker	59
Retrigger	81
Rise Offset	53
Roll Off	73
Roll Off Factor	49
Running	55
Running - Trigger	78

S

Sample Clock Mode	61
Save Bluetooth Settings	16
Save Configuration	16
Scan Interval	31
Scan Repetition Mode	24
Scan Window	31
Scanner	28
Scanner's device address	41
SCO	20
Select Data Channel Index	34
Selected data channel	34
SEQN Start Value	23
Sequence Length	20, 28, 68
Sequence Length (ARB)	66
Session Key iD(hex)	40
Set Synchronization Settings	61
Set to default	65
Set to Default	16
Set to Default (Dirty Transmitter Test)	46
Show Data Channel Mapping Table	40

Show Data Channel Table	40
Signal Duration	55
Signal Duration - Trigger	79
Signal Duration Unit - Trigger	55, 79
Simulation Mode	21
Slave	28
Slave Latency	39
Sleep Clock Accuracy	43
SN Start Value	37
Standard settings	65
Standard Settings	16
Start Phase	47
State	16, 68
State (Long Table)	47
State (Short Table)	47
Stop Signal Generation	19, 55
Stopped	55
Sub Version Number	44
Symbol Rate Variation	51
Symbol Timing Error (DTT)	48
Sync. Output to External Trigger	56
Synchronization mode	60
Synchronization Mode - TDSdCdma	88

T

Test Packet Configuration	46
Test Packet Interval	44
Test Packet Payload Length	45
Test Packet Payload Type	45
Transmit window offset	32, 42
Transmit window size	32, 42
Transport Mode	20
Transport Mode - ACL+EDR	20
Transport Mode - eSCO+EDR	20
Transport Mode - SCO	20
Trigger	19, 56
Trigger Delay	58
Trigger Delay External	80
Trigger Delay Other Baseband	78
Trigger Inhibit	58
Trigger Inhibit External	81
Trigger Inhibit Other Baseband	78
Trigger Mode	81
Armed	54
Auto	54
Retrigger	54
Single	54
Trigger Source	56

U

UAP	22
Unencrypted	33
Unknown Type(hex)	41
User Marker / AUX I/O Settings	62

V

Version	18, 69
Version Number	43

W

Waveform File	18
---------------------	----